CH641 Reference Manual

V1.3

https://wch-ic.com

Overview

The CH641 series is a dedicated MCU for PD wireless charging designed based on the QingKe RISC-V2A core. It supports 2-level interrupt nesting, has built-in PDPHY, BC interface, differential input current sampling and AC small signal amplification decoder, and supports USB PD and Type-C fast charging function, BC1.2 and other charging protocols, provides ADC, timer, serial port, I2C and other peripheral resources, and provides over-voltage protection and over-temperature protection.

This manual provides detailed usage information of the CH641 series microcontrollers for users' application development. It is applicable to products with different memory capacities, functional resources, and packages in the series. If there are any differences, special explanations will be made in the corresponding function chapters.

For the device characteristics of this product, please refer to the datasheet *CH641DS0*.

For information about the core, refer to *QingKeV2_Processor_Manual*.

RISC-V core version comparison overview

Features / Core versions	Instruction set	Hardware stack levels	Interrupt nesting levels	Fast interrupt channels	Flow line	Vector table model	Extended instruction	Debug interface
QingKe V2A	RV32EC	2	2	2	2	Address or command	Supported	1-wire

Abbreviated description of the bit attribute in the register:

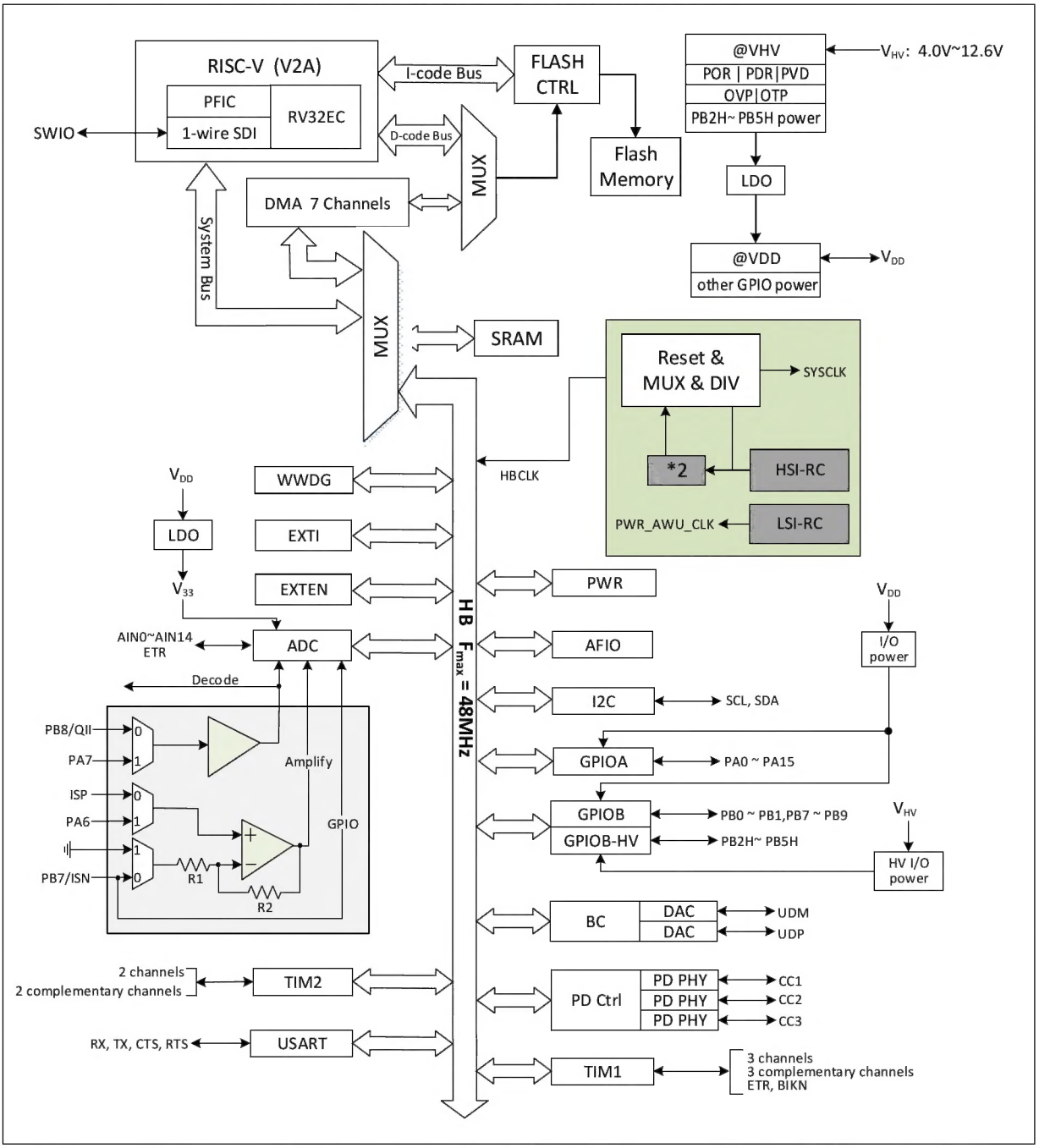
Register bit attributes	Property description
RF	Read-only attribute, read a fixed value.
RO	Read-only attribute, changed by hardware.
RZ	Read-only attribute, auto bit clear 0 after read operation.
WO	Write only attribute (not readable, read value uncertain)
WA	Write-only attribute, writable in Safe mode.
WZ	Write only attribute, auto bit clear 0 after write operation.
RW	Readable and writable.
RWA	Readable, writable in Safe mode.
RW1	Readable, write 1 valid, write 0 invalid.
RW0	Readable, write 0 valid, write 1 invalid.
RW1T	Readable, write 0 invalid, write 1 flipped.

Chapter 1 Memory and Bus Architecture

1.1 Bus Architecture

CH641 is based on the RISC-V instruction set, and its architecture realizes the interaction of the core, arbitration unit, DMA module, SRAM memory and other parts through multiple buses. The design integrates a general-purpose DMA controller to reduce the load on the CPU and improve the access efficiency, while at the same time, it also has a data protection mechanism, automatic clock switching protection and other measures to increase the stability of the system. Its system block diagram is shown in Figure 1-1.

Figure 1-1 CH641 system block diagram



The system is equipped with: General-purpose DMA controller to reduce the CPU load and improve efficiency;

clock tree hierarchy management to reduce the total power consumption of peripherals, as well as data protection mechanisms, clock security system protection mechanisms and other measures to increase system stability.

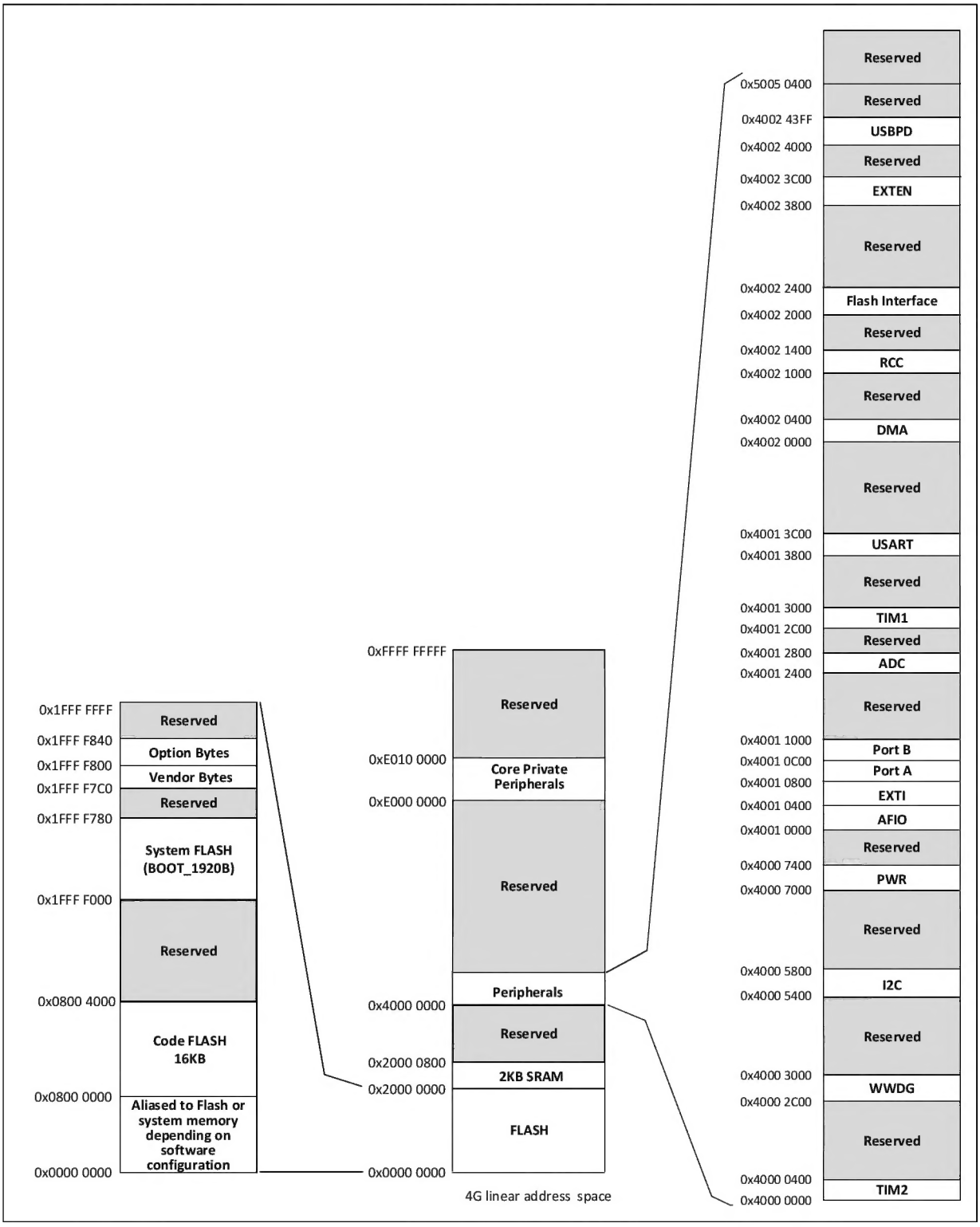
- The instruction bus (I-Code) connects the core to the FLASH instruction interface and prefetching is done on this bus.
- The data bus (D-Code) connects the core to the FLASH data interface for constant loading and debugging.
- The system bus connects the core to the bus matrix and is used to coordinate accesses to the core, DMA, SRAM and peripherals.
- The DMA bus is responsible for the DMA of the HB master interface connected to the bus matrix, which is accessed by FLASH data, SRAM and peripherals.
- The bus matrix is responsible for the access coordination between the system bus, data bus, DMA bus, SRAM and HB bridge.

1.2 Memory Map

The CH641 family contains program memory, data memory, core registers, peripheral registers, and more, all addressed in a 4GB linear space.

System storage stores data in small-end format, i.e., low bytes are stored at the low address and high bytes are stored at the high address.

Figure 1-2 Storage image



1.2.1 Memory Allocation

Built-in 2KB SRAM, starting address 0x20000000, supports byte, half-word (2 bytes), and full-word (4 bytes) access.

Built-in up to 16KB program Flash memory (CodeFlash) for storing user applications.

Built-in 1920-byte SystemFlash, or BOOT area, is used for system bootloader storage (factory-cured bootloader), and can also be used for user application program and constant data storage after setting the user configuration word START_MODE bit to 0 to turn off the BOOT (ld segmented link file required).

Built-in 64B space for vendor configuration word storage, factory-cured and unmodifiable by users.

Built-in 64B space for user option byte storage.

Chapter 2 Power Control (PWR)

2.1 Overview

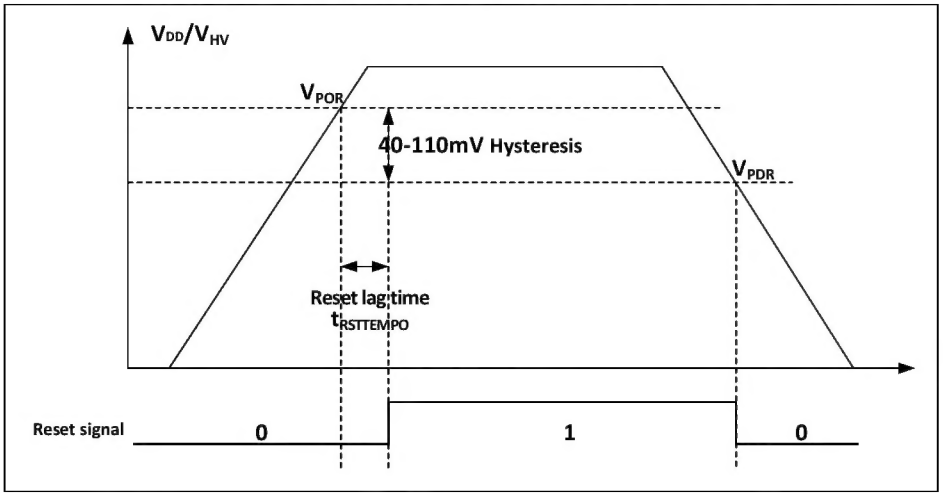
The system operating voltage V_{HV} range is 4.0 ~ 12.6V.

2.2 Power Management

2.2.1 Power-on Reset and Power-down Reset

The system integrated a power-on reset POR and a power-down reset PDR circuit. When the chip supply voltage V_{HV} or V_{DD} falls below the corresponding threshold voltage, the system is reset by the relevant circuit, and no additional external reset circuit is required. Please refer to the corresponding datasheet for the parameters of the power-on threshold voltage V_{POR} and the power-down threshold voltage V_{PDR}.

Figure 2-1 Schematic diagram of the operation of POR and PDR



2.2.2 Programmable Voltage Detector

The programmable voltage detector, PVD, is mainly used to monitor the change of the main power supply of the system and compare it with the threshold voltage set by PLS[2:0] of the power control register PWR_CTLR, and with the external interrupt register (EXTI) setting, it can generate relevant interrupts to notify the system in time for pre-power down operations such as data saving.

The specific configuration is as follows.

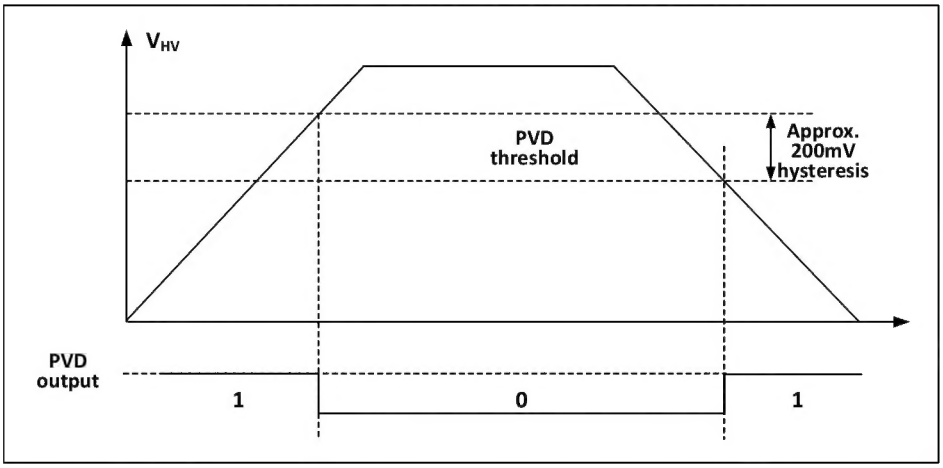
1) Set the PLS[1:0] field of the PWR_CTLR register and select the voltage threshold to be monitored.

2) Optional interrupt handling. the PVD function is internally connected to the rising/falling edge trigger setting of line 16 of the EXTI module, turning on this interrupt (configuring EXTI) will generate a PVD interrupt when V_{HV} drops below the PVD threshold or rises above the PVD threshold.

3) Set the PVDE bit of PWR_CTLR register to enable PVD function.

4) Read PVD0 bit of PWR_CSR status register can obtain the relationship between the current system main power supply and PLS[2:0] setting threshold, and perform corresponding soft processing. When the VDD voltage is higher than the threshold set by PLS[2:0], the PVD0 position is 0; When the VDD voltage is lower than the PLS[2:0] set threshold, the PVD0 bit is set.

Figure 2-2 Schematic diagram of PVD operation



2.2.3 Overvoltage Reset

The system continuously monitors the V_{HV} supply voltage internally and generates a reset when the V_{HV} voltage is higher than V_{OVP}.

2.3 Low-power Modes

After a system reset, the microcontroller is in a normal operating state (Run mode), where system power can be saved by reducing the system main frequency or turning off the unused peripheral clock or reducing the operating peripheral clock. If the system does not need to work, you can set the system to enter low-power mode and let the system jump out of this state by specific events.

Microcontrollers currently offer 2 low-power modes, divided in terms of operating differences between processors, peripherals, voltage regulators, etc.

- Sleep mode: The core stops running and all peripherals (including core private peripherals) are still running.
- Standby mode: Stops all clocks and the system continues to run after waking up.

Table 2-1 Low-power mode list

Mode	Entry	Wake-up source	Effect on clock	Voltage regulator
Sleep	WFI	Any interrupt	Core clock OFF, no effect on other clocks	Normal
	WFE	Wake-up event		
Standy	Set SLEEPDEEP to 1 Set PDDS to1 WFI or WFE	Any external interrupt/event (EXTI signal), external reset signal on RST, USB PD wakeup signal, PVD output, AWU auto-wakeup, the EXTI signal includes one of the 25 external I/O ports.	Disable HIS, PLL and peripheral clock	Off

Note: The SLEEPDEEP bit belongs to the core private peripheral control bit, CH641 refers to the PFIC_SCTLR register.

2.3.1 Low-power Configuration Options

- WFI and WFE

WFI: The microcontroller is woken up by an interrupt source with interrupt controller response, and the interrupt service function will be executed first after the system wakes up (except for microcontroller reset).

WFE: The wakeup event triggers the microcontroller to exit low-power mode. Wake-up events include:

1) Configure an external or internal EXTI line to event mode, when no interrupt controller needs to be configured.
2) Or configure an interrupt source, equivalent to a WFI wakeup, where the system prioritizes the execution of the interrupt service function.
3) Or configure the SEVONPEND bit to turn on peripheral interrupt enable, but not interrupt enable in the interrupt controller, and the interrupt pending bit needs to be cleared after the system wakes up.

- SLEEPONEXIT

  Enable: After executing the WFI or WFE instruction, the microcontroller ensures that all pending interrupt services are exited and then enters low-power mode.

  Not enabled: The microcontroller enters low-power mode immediately after executing the WFI or WFE command.
- SEVONPEND

  Enable: All interrupts or wake-up events can wake up the low-power consumption entered by executing WFE.

  Not enabled: Only interrupts or wake-up events enabled in the interrupt controller can wake up the low-power consumption entered by executing WFE.

2.3.2 Sleep Mode (SLEEP)

In this mode, all I/O pins keep their state in Run mode and all peripheral clocks are normal, so try to turn off useless peripheral clocks before entering Sleep mode to reduce low-power consumption. This mode takes the shortest time to wake up.

Enter: Configure core register control bit SLEEPDEEP=0, power control register PDDS=0, execute WFI or WFE, optionally SEVONPEND and SLEEPONEXIT.

Exit: Arbitrary interrupt or wakeup event.

2.3.3 Standby Mode (STANDBY)

Standby mode is based on the core's deep sleep mode (SLEEPDEEP), combined with the clock control mechanism of the peripherals, and allows the voltage regulator to operate in a lower power consumption state. In this mode, the high-frequency clock (HSI/PLL) domain is turned off, the SRAM and register contents are maintained, and the IO pin status is maintained. The system can continue to run after waking up in this mode, with HSI as the default system clock.

If flash programming is in progress, the system does not enter standby mode until the memory access is complete.

Modules that can work in Standby mode: low-frequency clock (LSI).

Enter: Configure the core register control bit SLEEPDEEP=1, PDDS=1 in the power control register, and execute WFI or WFE, optionally SEVONPEND and SLEEPONEXIT.

Exit: Any external interrupt/event (EXTI signal), external reset signal on RST, USBPD wake-up signal, PVD output, AWU automatic wake-up, where the EXTI signal includes one of the 25 external I/O ports

2.3.4 Auto-wakeup (AWU)

Automatic wake-up without external interrupt can be realized. Periodic wake-up from low-power mode is possible by programming the time base.

The selectable internal low-frequency clock oscillator LSI divides by 60000 as the automatic wake-up counting time base.

2.4 Register Description

Table 2-2 PWR-related registers list

Name	Access address	Description	Reset value
R32_PWR_CTLR	0x40007000	Power control register	0x00000000
R32_PWR_CSR	0x40007004	Power control/status register	0x00000000
R32_PWR_AWUCSR	0x40007008	Automatic wake-up control status register	0x00000000
R32_PWR_AWUWR	0x4000700C	Automatic wake-up window comparison value register	0x00000000

2.4.1 Power Control Register (PWR_CTLR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								PLS[2:0]			PVDE	Reserved		PDDS	Reserved

Bit	Name	Access	Description	Reset value
[31:8]	Reserved	RO	Reserved	0
[7:5]	PLS[2:0]	RW	PVD voltage monitoring threshold setting. See the Electrical Characteristics section of the datasheet for detailed descriptions. 000: Rising edge 3.16V / falling edge 2.94V; 001: Rising edge 3.38V / falling edge 3.12V; 010: Rising edge 3.61V / falling edge 3.32V; 011: Rising edge 3.82V / falling edge 3.51V; 100: Rising edge 4.06V / falling edge 3.70V; 101: Rising edge 4.28V / falling edge 3.92V; 110: Rising edge 4.45V / falling edge 4.09V; 111: Rising edge 4.63V / falling edge 4.28V.	0
4	PVDE	RW	Power supply voltage monitoring function enable flag. 1: Enable power supply voltage monitoring function; 0: Disable power supply voltage monitoring function.	0
[3:2]	Reserved	RO	Reserved	0
1	PDDS	RW	In the power-down deep sleep scenario, the standby/sleep mode selection bit. 1: Entry Standby mode; 0: Entry Sleep mode	0
0	Reserved	RO	Reserved	0

2.4.2 Power Control/Status Register (PWR_CSR)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved													PVD0	Reserved	

Bit	Name	Access	Description	Reset value
[31:3]	Reserved	RO	Reserved	0
2	PVD0	RO	PVD output status flag bit. This bit is valid when PVDE=1 in PWR_CTLR register. 1: V_{HV} is below the PVD threshold set by PLS[2:0]; 0: V_{HV} is above the PVD threshold set by PLS[2:0].	0
[1:0]	Reserved	RO	Reserved	0

2.4.3 Automatic Wake-up Control Status Register (PWR_AWUCSR)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														AWUEN	AWUCLR

Bit	Name	Access	Description	Reset value
[31:2]	Reserved	RO	Reserved	0
1	AWUEN	RW	Auto-wake up enable: 1: Turn on automatic wake-up, enable LSI clock; 0: Turn off automatic wake-up function and LSI, then clear AWUCLR, AWUWR and counter.	0
0	AWUCLR	WO	Automatic wake-up counter clear control bit: 1: Clear AWU internal counter (This position is automatically cleared by hardware after 1);	0

2.4.4 Automatic Wake-up Window Comparison Value Register (PWR_AWUWR)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												AWUUWR[3:0]			

Bit	Name	Access	Description	Reset value
[31:4]	Reserved	RO	Reserved	0
[3:0]	AWUWR[3:0]	RW	AWU window value; The AWU window value is equal to the input value of the AWU window value + 1; The AWU window value is used to compare with the incremental counter value and generate a wake-up signal when the counter value is equal to the window value.	0

Chapter 3 Reset and Clock Control (RCC)

The controller provides different forms of resets and configurable clock tree structures based on the division of power areas and peripheral power management considerations in the application. This section describes the scope of each clock in the system.

3.1 Main Features

- Multiple reset forms
- Multiple clock sources, bus clock management
- Independent management of each peripheral clock: Reset, On, Off
- Supports internal clock output

3.2 Reset

The controller provides 3 forms of reset: Power reset, system reset, overtemperature or overvoltage reset.

3.2.1 Power Reset

When a power reset occurs, it will reset all registers.

A power Reset is generated when the following event occurs:

- Power-up/power-down reset (POR/PDR)

3.2.2 System Reset

When a system reset occurs, it will reset the reset flag in addition to the control/status register RCC_RSTSCKR and all the registers. The source of the reset event is identified by looking at the reset status flag bit in the RCC_RSTSCKR register.

A system reset is generated when one of the following events occurs:

- Low signal on NRST pin (external reset)
- Window watchdog count termination (WWDG reset)
- Software reset (SW reset)
- USBPD reset
- Low-power management reset
- OVP overvoltage reset

Window watchdog reset: triggered by the window watchdog external timer count cycle overflow, see the corresponding chapter for detailed description.

Software reset: The CH641 resets the system via RSTSYS position 1 of the Interrupt Configuration Register PFIC_CFGR in the Programmable Interrupt Controller PFIC or SYSRST position 1 of the Configuration Register PFIC_SCTLR in the System Cabinet, refer to the corresponding section.

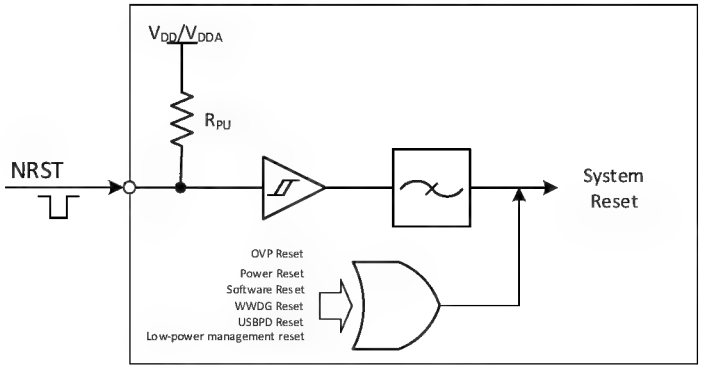
USBPD reset: CH641 supports the reset generated by USB PD signal frame Hard Reset when PD_RST_EN is 1, and also supports the reset generated by signal frame Cable Reset if IE_RX_RESET is also 1. USB PD does not have a reset flag, but the reset effect is the same as the software reset.

Low Power Management reset: Standby mode reset is enabled by setting the STANDBY_RST bit in the user select byte. After executing the process of entering standby mode at this time, a system reset will be executed instead of

entering Standby mode.

OVP overvoltage reset: triggered when the V_{HV} voltage is higher than V_{OVP}.

Figure 3-1 System reset structure



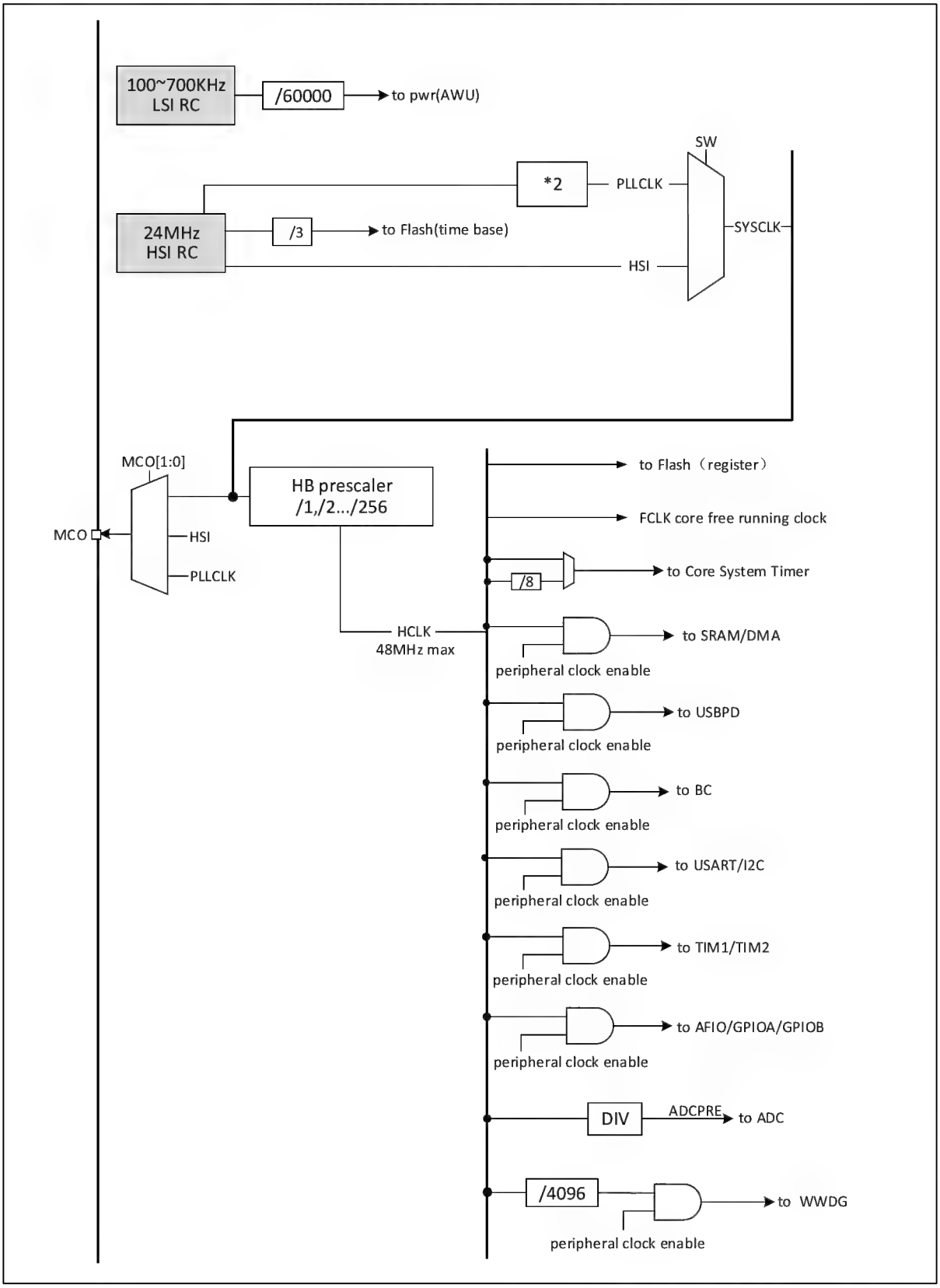
3.2.3 Overtemperature Reset

CH641 has a built-in overtemperature protection (OTP)module, which will forcefully reset the MCU when the chip temperature is too high.

3.3 Clock

3.3.1 System Clock Structure

Figure 3-2 CH641 clock tree block diagram



3.3.2 High-speed Internal Clock (HSI)

HSI is a high-speed clock signal generated by the 24MHz RC oscillator inside the system. The HSIRC oscillator provides the system clock without the need for any external components. Its startup time is very short. HSI is enabled and disabled by setting the HSION bit in the RCC_CTLR register, and the HSIRDY bit indicates whether the HSIRC oscillator is stable. By default, HSION and HSIRDY are set to 1 by the system (it is recommended not to turn them off). If the HSIRDYIE bit of the RCC_INTR register is set, the corresponding interrupt will be generated.

- Factory calibration: The difference of manufacturing process will cause different RC oscillation frequency for each chip, so HSI calibration is performed for each chip before it is shipped. After system reset, the factory calibration value is loaded into HSICAL[7:0] of the RCC_CTLR register.
- User tuning: Based on different voltages or ambient temperatures, the application can adjust the HSI frequency by using the HSITRIM[4:0] bits in the RCC_CTLR register.

3.3.3 Low-speed Internal Clock (LSI)

LSI is a low-speed clock signal generated by the RC oscillator inside the system. It can remain running in standby mode to provide a clock reference for waking up the unit. Further information can be found in the Electrical Characteristics section of the datasheet.

3.3.4 PLL Clock

The PLL clock is turned on and off by setting the PLLON bit in the RCC_CTLR register. The PLLRDY bit indicates whether the PLL clock is stable or not, and the hardware sends the clock into the system only after PLL position 1. If the PLLRDYIE bit in the RCC_INTR register is set, a corresponding interrupt is generated.
PLL clock source: HSI clock input.

3.3.5 Bus/Peripheral Clock

3.3.5.1 System Clock (SYSCLK)

Configure the system clock source by configuring the SW[1:0] bits of the RCC_CFGR0 register. SWS[1:0] indicates the current system clock source.

- HSI as system clock
- PLL clock as system clock

After the controller is reset, the default HSI clock is selected as the system clock source. Switching between clock sources must not occur until the target clock source is ready.

3.3.5.2 HB Bus Peripheral Clock (HCLK)

The HB bus clocks can be configured by configuring the HPRE[3:0] bits of the RCC_CFGR0 register. The bus clock determines the peripheral interface access clock reference that is mounted below them. Applications can adjust different values to reduce the power consumption when some of the peripherals are operating.
The various bits in the RCC_APB1PRSTR and RCC_APB2PRSTR registers can reset the different peripheral modules to their initial state.
Each bit in the RCC_AHBPCENR, RCC_APB1PCENR, and RCC_APB2PCENR registers can be used to individually turn on or off the communication clock interface for different peripheral modules. When using a peripheral, you first need to turn on its clock enable bit in order to access its registers.

3.3.5.3 Microcontroller Clock Output (MCO)

The microcontroller allows outputting clock signals to the MCO pins. The following 4 clock signals can be selected as MCO clock outputs by configuring the alternate push-pull output mode in the corresponding GPIO port registers by configuring the MCO[2:0] bits of the RCC_CFGR0 register.

- System clock (SYSCLK) output
- HSI clock output
- PLL clock output

3.4 Register Description

Table 3-1 RCC-related registers list

Name	Access address	Description	Reset value
R32_RCC_CTLR	0x40021000	Clock control register	0x0000xx83
R32_RCC_CFGR0	0x40021004	Clock configuration register 0	0x00000020
R32_RCC_INTR	0x40021008	Clock interrupt register	0x00000000
R32_RCC_APB2PRSTR	0x4002100C	PB2 peripheral reset register	0x00000000
R32_RCC_APB1PRSTR	0x40021010	PB1 peripheral reset register	0x00000000
R32_RCC_AHBPCENR	0x40021014	HB peripheral clock enable register	0x00000004
R32_RCC_APB2PCENR	0x40021018	PB2 peripheral clock enable register	0x00000000
R32_RCC_APB1PCENR	0x4002101C	PB1 peripheral clock enable register	0x00000000
R32_RCC_RSTSCKR	0x40021024	Control/status register	0x0C000000

3.4.1 Clock Control Register (RCC_CTLR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						PLL RDY	PLLON	Reserved							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSICAL[7:0]								HSITRIM[4:0]					Reserved	HSI RDY	HSION

Bit	Name	Access	Description	Reset value
[31:26]	Reserved	RO	Reserved	0
25	PLLRDY	RO	PLL clock ready lock flag bit 1: PLL clock lock; 0: PLL clock not locked.	0
24	PLLON	RW	PLL clock enable control bit 1: Enable PLL clock; 0: Disable PLL clock. *Note: After entering standby low-power mode, this bit is cleared by hardware.*	0
[23:16]	Reserved	RO	Reserved	0
[15:8]	HSICAL[7:0]	RO	The internal high-speed clock calibration value is automatically initialized when the system starts.	x
[7:3]	HSITRIM[4:0]	RW	Internal high-speed clock adjustment value. The user can input an adjustment value to be superimposed on the HSICAL[7:0] value to adjust the frequency of the internal HSIRC oscillator according to changes in voltage and temperature. The default value is 16, which allows you to adjust the HSI to 24MHz ±1%; the change in HSICAL is adjusted about	10000b

			60KHz per step.	
2	Reserved	RO	Reserved	0
1	HSIRDY	RO	Internal high-speed clock (24MHz) stable ready flag (set by hardware). 1: The internal high-speed clock (24MHz) is stable; 0: The internal high-speed clock (24MHz) is not stable. *Note: After the HSION bit is cleared to 0, it takes 6 HSI cycles for the bit to be cleared to 0.*	1
0	HSION	RW	Internal high-speed clock (24MHz) enable control bit. 1: Enable HSI oscillator; 0: Disable HSI oscillator. *Note: When returning from standby mode, this bit is set by hardware to start the internal 24MHz RC oscillator.*	1

3.4.2 Clock Configuration Register0 (RCC_CFGR0)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved					MCO[2:0]			Reserved							ADC PRE[5]

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCPRE[4:0]					Reserved			HPRE[3:0]				SWS[1:0]		SW[1:0]	

Bit	Name	Access	Description	Reset value
[31:27]	Reserved	RO	Reserved	0
[26:24]	MCO[2:0]	RW	Microcontroller MCO pin clock output control 0xx: No clock output; 100: SYSCLK output; 101: Internal 24MHz RC oscillator clock (HSI) output; 110: Reserved; 111：PLL clock output.	0
[23:17]	Reserved	RO	Reserved	0
[16:11]	ADCPRE[5:0]	RW	AHB clock source prescaler control: 0000xx: HBCLK divided by 2 as ADC clock; 0010xx: HBCLK divided by 4 as ADC clock; 0100xx: HBCLK divided by 6 as ADC clock; 0110xx: HBCLK divided by 8 as ADC clock; 1000xx: HBCLK divided by 12 as ADC clock; 1010xx: HBCLK divided by 16 as ADC clock; 1100xx: HBCLK divided by 24 as ADC clock; 1110xx: HBCLK divided by 48 as ADC clock; 000100: HBCLK divided by 4 as ADC clock; 001100: HBCLK divided by 8 as ADC clock; 010100: HBCLK divided by 12 as ADC clock; 011100: HBCLK divided by 16 as ADC clock; 100100: HBCLK divided by 24 as ADC clock; 101100: HBCLK divided by 32 as ADC clock; 110100: HBCLK divided by 48 as ADC clock; 111100: HBCLK divided by 96 as ADC clock; 000101: HBCLK divided by 8 as ADC clock; 001101: HBCLK divided by 16 as ADC clock; 010101: HBCLK divided by 24 as ADC clock; 011101: HBCLK divided by 32 as ADC clock;	0

			100101: HBCLK divided by 48 as ADC clock; 101101: HBCLK divided by 64 as ADC clock; 110101: HBCLK divided by 96 as ADC clock; 111101: HBCLK divided by 192 as ADC clock; 000110: HBCLK divided by 16 as ADC clock;; 001110: HBCLK divided by 32 as ADC clock; 010110: HBCLK divided by 48 as ADC clock; 011110: HBCLK divided by 64 as ADC clock; 100110: HBCLK divided by 96 as ADC clock; 101110: HBCLK divided by 128 as ADC clock; 110110: HBCLK divided by 192 as ADC clock; 111110: HBCLK divided by 384 as ADC clock; 000111: HBCLK divided by 32 as ADC clock; 001111: HBCLK divided by 64 as ADC clock; 010111: HBCLK divided by 96 as ADC clock; 011111: HBCLK divided by 128 as ADC clock; 100111: HBCLK divided by 192 as ADC clock; 101111: HBCLK divided by 256 as ADC clock; 110111: HBCLK divided by 384 as ADC clock; 111111: HBCLK divided by 768 as ADC clock; *Note: ADC clock should not exceed 24MHz maximum.*	
[10:8]	Reserved	RW	Reserved	0
[7:4]	HPRE[3:0]	RW	HB clock source prescaler control. 0000: Prescaler off. 0001: SYSCLK divided by 2. 0010: SYSCLK divided by 3. 0011: SYSCLK divided by 4. 0100: SYSCLK divided by 5. 0101: SYSCLK divided by 6. 0110: SYSCLK divided by 7. 0111: SYSCLK divided by 8. 1000: SYSCLK divided by 2. 1001: SYSCLK divided by 4. 1010: SYSCLK divided by 8. 1011: SYSCLK divided by 16. 1100: SYSCLK divided by 32. 1101: SYSCLK divided by 64. 1110: SYSCLK divided by 128. 1111: SYSCLK divided by 256.	0010b
[3:2]	SWS[1:0]	RO	System clock (SYSCLK) status (hardware set). 00: The system clock source is HSI; 10: The system clock source is PLL; Other: Not available.	0
[1:0]	SW[1:0]	RW	Select the system clock source. 00: HSI as the system clock; 10: PLL output as the system clock; Other: Not available.	0

3.4.3 Clock Interrupt Register (RCC_INTR)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved											PLL RDY C	Reserved			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			PLLRDYIE	Reserved							PLLRDYF	Reserved			

Bit	Name	Access	Description	Reset value
[31:21]	Reserved	RO	Reserved	0
20	PLLRDYC	WO	Clear the PLL ready interrupt flag bit. 1: Clear PLLRDYF interrupt flag; 0: No action.	0
[19:13]	Reserved	RO	Reserved	0
12	PLLRDYIE	RW	PLL ready interrupt enable bit. 1: Enable PLL ready interrupt; 0: Disable PLL ready interrupt.	0
[11:5]	Reserved	RO	Reserved	0
4	PLLRDYF	RO	PLL clock ready lock interrupt flag. 1: The PLL clock lock generates an interrupt; 0: No PLL clock lock interrupt. Hardware set, software write PLLRDYC bit 1 clear.	0
[3:0]	Reserved	RO	Reserved	0

3.4.4 PB2 Peripheral Reset Register (RCC_APB2PRSTR)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	USART1RST	Reserved		TIM1RST	Reserved	ADC1RST	Reserved					IOPBRST	IOPARST	Reserved	AFIORST

Bit	Name	Access	Description	Reset value
[31:15]	Reserved	RO	Reserved	0
14	USART1RST	RW	USART1 interface reset control. 1: Reset module; 0: No effect.	0
[13:12]	Reserved	RO	Reserved	0
11	TIM1RST	RW	TIM1 module reset control. 1: Reset module; 0: No effect.	0
10	Reserved	RO	Reserved	0
9	ADC1RST	RW	ADC1 module reset control. 1: Reset module; 0: No effect.	0
[8:4]	Reserved	RO	Reserved	0
3	IOPBRST	RW	IO PB port module reset control. 1: Reset module; 0: No effect.	0
2	IOPARST	RW	IO PA port module reset control. 1: Reset module; 0: No effect.	0
1	Reserved	RO	Reserved	0
0	AFIORST	RW	IO auxiliary function module reset control. 1: Reset module; 0: No effect.	0

3.4.5 PB1 Peripheral Reset Register (RCC_APB1PRSTR)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			PWR RST	Reserved						I2C1 RST	Reserved				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				WW DG RST	Reserved										TIM 2 RST

Bit	Name	Access	Description	Reset value
[31:29]	Reserved	RO	Reserved	0
28	PWRRST	RW	Power interface module reset control. 1: Reset module; 0: No effect.	0
[27:22]	Reserved	RO	Reserved	0
21	I2C1RST	RW	I2C1 interface reset control. 1: Reset module; 0: No effect.	0
[20:12]	Reserved	RO	Reserved	0
11	WWDGRST	RW	WWDG reset control. 1: Reset module; 0: No effect.	0
[10:1]	Reserved	RO	Reserved	0
0	TIM2RST	RW	Timer 2 module reset control. 1: Reset module; 0: No effect.	0

3.4.6 HB Peripheral Clock Enable Register (RCC_AHBPCENR)

Offset address: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								USBP DEN	Reserved				SRA M EN	Reser ved	DMA 1 EN

Bit	Name	Access	Description	Reset value
[31:8]	Reserved	RO	Reserved	0
7	USBPDEN	RW	USBPD module clock enable: 1: Module clock on; 0: Module clock off.	0
[6:3]	Reserved	RO	Reserved	0
2	SRAMEN	RW	SRAM interface module clock enable bit: 1: SRAM interface module clock on in Sleep mode; 0: SRAM interface module clock off in Sleep mode.	1
1	Reserved	RO	Reserved	0
0	DMA1EN	RW	DMA1 module clock enable bit: 1: Module clock on; 0: Module clock off.	0

3.4.7 PB2 Peripheral Clock Enable Register (RCC_APB2PCENR)

Offset address: 0x18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	USART1 EN	Reserved		TIM1 EN	Reserved	ADC1 EN	Reserved					IOPB EN	IOPA EN	Reserved	AFIO EN

Bit	Name	Access	Description	Reset value
[31:15]	Reserved	RO	Reserved	0
14	USART1EN	RW	USART1 interface clock enable bit: 1: Module clock is on; 0: Module clock is off.	0
[13:12]	Reserved	RO	Reserved	0
11	TIM1EN	RW	TIM1 module clock enable bit: 1: Module clock is on; 0: Module clock is off.	0
10	Reserved	RO	Reserved	0
9	ADC1EN	RW	ADC1 module clock enable bit: 1: Module clock is on; 0: Module clock is off.	0
[8:4]	Reserved	RO	Reserved	0
3	IOPBEN	RW	PB port module clock enable bit for I/O: 1: Module clock is on; 0: Module clock is off.	0
2	IOPAEN	RW	PA port module clock enable bit for I/O: 1: Module clock is on; 0: Module clock is off.	0
1	Reserved	RO	Reserved	0
0	AFIOEN	RW	I/O auxiliary function module clock enable bit: 1: Module clock is on; 0: Module clock is off.	0

3.4.8 PB1 Peripheral Clock Enable Register (RCC_APB1PCENR)

Offset address: 0x1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			PWR EN	Reserved						I2C1 EN	Reserved				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				WWDG EN	Reserved										TIM2 EN

Bit	Name	Access	Description	Reset value
[31:29]	Reserved	RO	Reserved	0
28	PWREN	RW	Power interface module clock enable bit: 1: Module clock is on; 0: Module clock is off.	0
[27:22]	Reserved	RO	Reserved	0
21	I2C1EN	RW	I2C1 interface clock enable bit: 1: Module clock is on; 0: Module clock is off.	0
[20:12]	Reserved	RO	Reserved	0
11	WWDGEN	RW	Window watchdog clock enable bit: 1: Module clock is on; 0: Module clock is off.	0
[10:1]	Reserved	RO	Reserved	0
0	TIM2EN	RW	Timer 2 module clock enable bit: 1: Module clock is on; 0: Module clock is off.	0

3.4.9 Control/Status Register (RCC_RSTSCKR)

Offset address: 0x24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPWRRSTF	WWDGRSTF	Reserved	SFTRSTF	PORRSTF	PINRSTF	USBPDRSTF	RMVF	Reserved							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															

Bit	Name	Access	Description	Reset value
31	LPWRRSTF	RO	Low-power reset flag: 1: Occurrence of low-power resets. 0: No low-power reset occurs. Set to 1 by hardware when a low-power management reset occurs; cleared by software writing of the RMVF bit.	0
30	WWDGRSTF	RO	Window watchdog reset flag: 1: Occurrence of a window watchdog reset. 0: No window watchdog reset occurs. Set to 1 by hardware when a window watchdog reset occurs; cleared by software writing of the RMVF bit.	0
29	Reserved	RO	Reserved	0
28	SFTRSTF	RO	Software reset flag: 1: Software reset occurs. 0: No software reset occurs. Set to 1 by hardware when a software reset occurs; software write RMVF bit cleared.	0
27	PORRSTF	RO	Power-up/power-down reset flag: 1: Power-up/power-down reset occurs. 0: No power-up/power-down reset occurs. Set to 1 by hardware when power-up/power-down reset occurs; cleared by software writing of RMVF bit.	1
26	PINRSTF	RO	External manual reset (NRST pin) flag: 1: Occurrence of NRST pin reset. 0: No NRST pin reset occurs. Set to 1 by hardware when NRST pin reset occurs; cleared by software writing of RMVF bit.	0
25	USBPDRSTF	RO	USBPD reset flag: 1: Occurrence of USB PD pin reset. 0: No USB PD pin reset occurs. Set to 1 by hardware when USB PD reset occurs; cleared by software writing of RMVF bit.	0
24	RMVF	RW	Clear reset flag control: 1: Clear the reset flag. 0: No effect.	0
[23:0]	Reserved	RO	Reserved	0

Note: Except for the reset flag which can only be cleared by power-on reset, others are cleared by system Reset.

Chapter 4 Window Watchdog (WWDG)

Window Watchdog is generally used to monitor system operation for software faults such as external disturbances, unforeseen logic errors, and other conditions. It requires a counter refresh (dog feeding) within a specific window time (with upper and lower limits), otherwise earlier or later than this window time the watchdog circuit will generate a system Reset.

4.1 Main Features

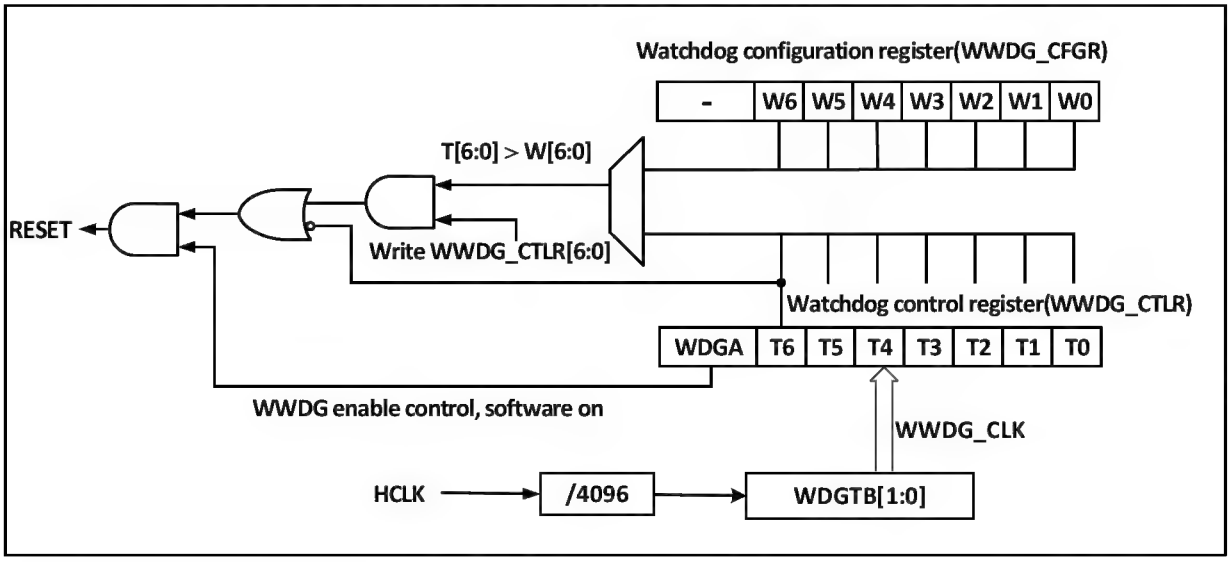
- Programmable 7-bit down-counter
- Biconditional reset: the down-counter value is less than 0x40, or the counter value is reloaded outside the window time
- Wake Up Early Notification (EWI) function for timely dog feeding action to prevent system Reset

4.2 Function Description

4.2.1 Principle and Application

The window watchdog operation is based on a 7-bit down-counter, which is mounted under the HB bus and counts the dividing frequency of the time base WWDG_CLK source (HCLK/4096) clock with the dividing factor set in the WDGTB[1:0] field in the configuration register WWDG_CFGR. The down-counter is in the free-running state, and the counter keeps counting down in a loop regardless of whether the watchdog function is on or not. As shown in Figure 4-1, the block diagram of the internal structure of the window watchdog.

Figure 4-1 Block diagram of Window Watchdog structure



- Enable Window Watchdog

After a system Reset, the watchdog is off. Setting the WDGA bit of the WWDG_CTLR register enables the watchdog, and then it cannot be turned off again unless a reset occurs.

Note: The watchdog function can be stopped indirectly by setting the RCC_APB1PCENR register to turn off the clock source of WWDG and suspend the WWDG_CLK count, or by setting the RCC_APB1PRSTR register to reset the WWDG module, which is equivalent to the role of reset.

- Watchdog Configuration

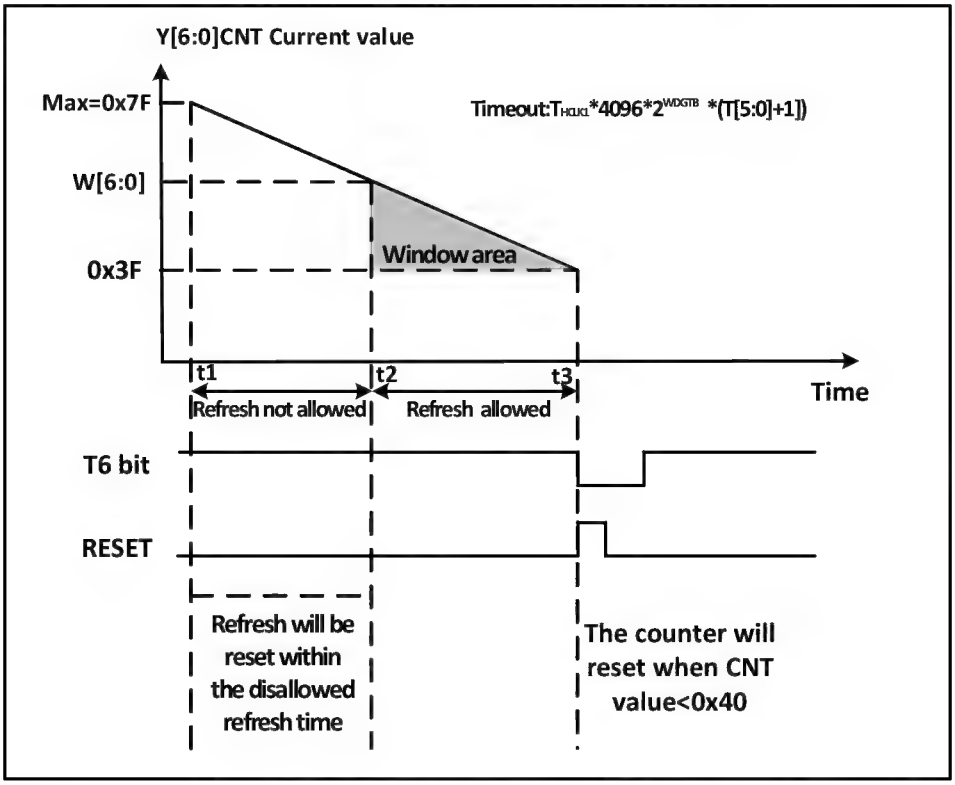
The watchdog is internally a 7-bit counter that runs in a continuous decreasing cycle and supports read and write access. To use the watchdog reset function, the following actions need to be performed.

1) Counting time base: via the WDGTB[1:0] bit field of the WWDG_CFGR register, note that the WWDG module clock of the RCC unit should be turned on.
2) Window counter: Set the W[6:0] bit field of WWDG_CFGR register, this counter is used by hardware as a comparison with the current counter, the value is configured by user software and will not change. It is used as the upper limit value of the window time.
3) Watchdog enable: WDG_CTLR register WDGA bit software set to 1, to turn on the watchdog function, you can system reset.
4) Feed the dog: i.e., refresh the current counter value and configure the T[6:0] bit field of the WWDG_CTLR register. This action needs to be executed within the periodic window time after the watchdog function is turned on, otherwise a watchdog reset action will occur.

- Dog feeding window time

As shown in Figure 4-2, the gray area is the monitoring window area of the window watchdog, whose upper time t2 corresponds to the point in time when the current counter value reaches the window value W[6:0]; its lower time t3 corresponds to the point in time when the current counter value reaches 0x3F. This area time t2<t<t3 can be fed with a dog operation (write T[6:0]) to refresh the current counter value.

Figure 4-2 Counting mode of Window Watchdog



- Watchdog reset

1) When the value of T[6:0] counter changes from 0x40 to 0x3F due to no timely dog feeding operation, a "window watchdog reset" will occur and a system reset will be generated. That is, the T6-bit is detected as 0 by the hardware and a system reset will occur.

   Note: The application can write T6-bit to 0 by software to achieve system Reset, which is equivalent to software

reset function.

2) When the counter refresh action is executed within the disallowed dog feeding time, i.e., the write T[6:0] bit field is operated within $t_1 \leq t \leq t_2$ time, a "window watchdog reset" will occur and a system Reset will be generated.

- Wake up in advance

To prevent the system Reset caused by not refreshing the counter in time, the watchdog module provides an early wakeup interrupt (EWI) notification. When the counter self-decreases to 0x40, an early wake-up signal is generated and the EWIF flag is set to 1. If the EWI bit is set, a window watchdog interrupt will be triggered at the same time. At this time, there is 1 counter clock cycle (self-decrement to 0x3F) before the hardware reset, and the application can perform the dog feeding operation instantly within this time.

4.2.2 Debug Mode

When the system enters Debug mode, the counter of WWDG can be configured by the debug module register to continue or stop.

4.3 Register Description

Table 4-1 WWDG-related registers list

Name	Access address	Description	Reset value
R16_WWDG_CTLR	0x40002C00	Control register	0x007F
R16_WWDG_CFGR	0x40002C04	Configuration Register	0x007F
R16_WWDG_STATR	0x40002C08	Status Register	0x0000

4.3.1 Control Register (WWDG_CTLR)

Offset address: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								WDGA	T[6:0]						

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
7	WDGA	RW1	Window watchdog reset enable bit. 1: Turn on the watchdog function (which generates a reset signal). 0: Disable the watchdog function. Software write 1 is on, but only allows hardware to clear 0 after reset.	0
[6:0]	T[6:0]	RW	The 7-bit self-decrement counter decrements by 1 every $4096*2^{WDGTB}$ HCLK cycles. A watchdog reset is generated when the counter decrements from 0x40 to 0x3F, i.e., when T6 jumps to 0.	1111111b

4.3.2 Configuration Register (WWDG_CFGR)

Offset address: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						EWI	WDGTB[1:0]		W[6:0]						

Bit	Name	Access	Description	Reset value
[15:10]	Reserved	RO	Reserved	0
9	EWI	RW1	Early wakeup interrupt enable bit. If this position is 1, an interrupt is generated when the counter value reaches 0x40. This bit can only be invited to 0 by hardware after a reset.	0
[8:7]	WDGTB[1:0]	RW	Window watchdog clock division selection. 00: Divided by 1, counting time base = HCLK/4096. 01: Divided by 2, counting time base = HCLK /4096/2. 10: Divided by 4, counting time base = HCLK /4096/4. 11: Divided by 8, counting time base = HCLK /4096/8.	0
[6:0]	W[6:0]	RW	Window watchdog 7-bit window value. Used to compare with the counter value. The feed dog operation can only be performed when the counter value is less than the window value and greater than 0x3F.	1111111b

4.3.3 Status Register (WWDG_STATR)

Offset address: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															EWIF

Bit	Name	Access	Description	Reset value
[15:1]	Reserved	WO	Reserved	0
0	EWIF	RW0	Wake up the interrupt flag bit early. When the counter reaches 0x40, this bit is set in hardware and must be cleared to 0 by software; the user setting is invalid. Even if the EWI is not set, this bit will still be set as usual when the event occurs.	0

Chapter 5 Interrupt and Events (PFIC)

The CH641 series has a built-in Programmable Fast Interrupt Controller (PFIC) that supports up to 255 interrupt vectors. The current system manages 25 peripheral interrupt channels and 4 core interrupt channels, the others are reserved.

5.1 Main Features

5.1.1 PFIC

- 25 peripheral interrupts, each interrupt request has independent trigger and mask control bits, with dedicated status bits
- Programmable multi-level interrupt nesting, maximum nesting depth 2 levels, hardware stack depth 2 levels
- Fast interrupt entry and exit mechanism, hardware automatic stacking and recovery
- Vector Table Free (VTF) interrupt response mechanism, 2-way programmable direct access to interrupt vector addresses

5.2 System Timer

- CH641 Series

The core comes with a 32-bit add counter (SysTick) that supports HCLK or HCLK/8 as a time base with high priority and can be used as a time reference after calibration.

5.3 Vector Table of Interrupts and Exceptions

Table 5-1 CH641 series vector table

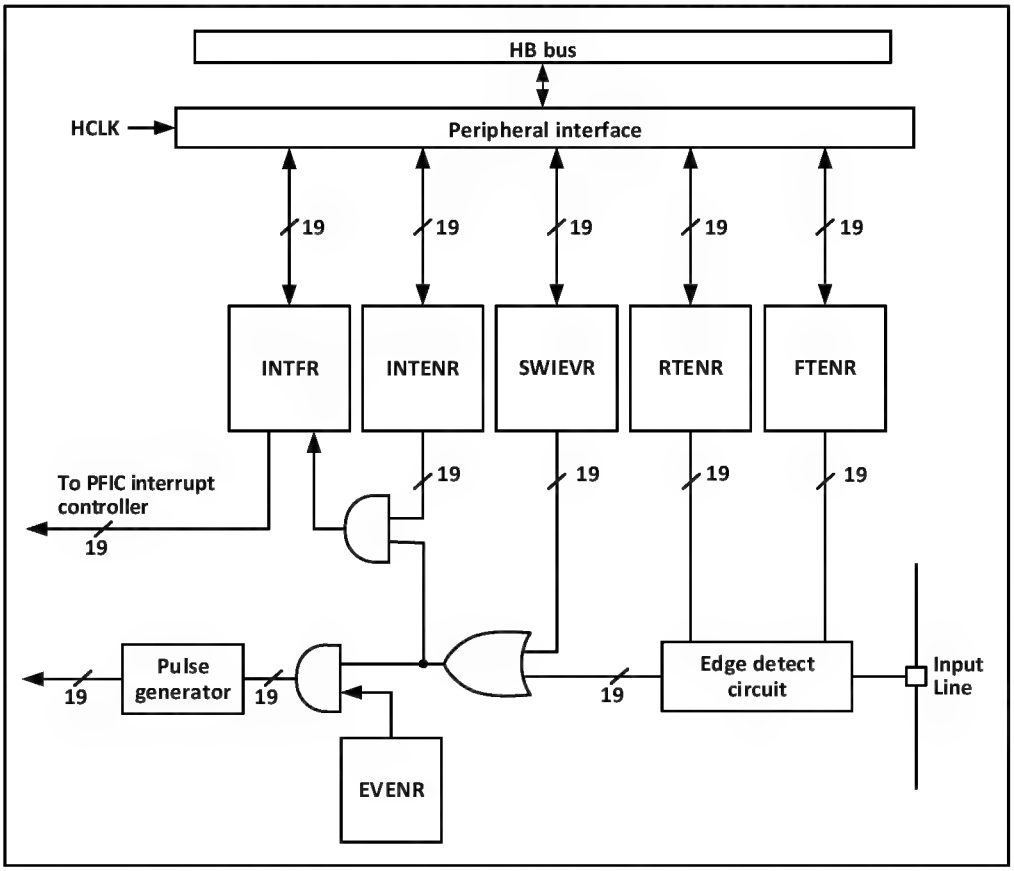
No.	Priority	Type	Name	Description	Entrance address
0	-	-	-	-	0x00000000
1	-	-	-	-	0x00000004
2	-2	-	-	-	0x00000008
3	-1	fixed	HardFault	Abnormal interrupts	0x0000000C
4-11	-	-	-	Reserved	0x00000010-0x0000002C
12	0	programmable	SysTick	System timer interrupt	0x00000030
13	-	-	-	Reserved	0x00000034
14	1	programmable	SW	Software interrupt	0x00000038
15	-	-	-	Reserved	0x0000003C
16	2	programmable	WWDG	Window timer interrupt	0x00000040
17	3	programmable	PVD	Supply voltage detection interrupt (EXTI)	0x00000044
18	4	programmable	FLASH	FLASH interrupt	0x00000048
19	5	programmable	RCC	Reset and clock control interrupt	0x0000004C
20	6	programmable	EXTI7_0	EXTI line 0-7 interrupt	0x00000050
21	7	programmable	AWU	Wake-up interrupt	0x00000054
22	8	programmable	DMA1_CH1	DMA1 channel 1 global interrupt	0x00000058
23	9	programmable	DMA1_CH2	DMA1 channel 2 global interrupt	0x0000005C
24	10	programmable	DMA1_CH3	DMA1 channel 3 global interrupt	0x00000060
25	11	programmable	DMA1_CH4	DMA1 channel 4 global interrupt	0x00000064
26	12	programmable	DMA1_CH5	DMA1 channel 5 global interrupt	0x00000068

27	13	programmable	DMA1_CH6	DMA1 channel 6 global interrupt	0x0000006C
28	14	programmable	DMA1_CH7	DMA1 channel 7 global interrupt	0x00000070
29	15	programmable	ADC	ADC global interrupt	0x00000074
30	16	programmable	I2C1_EV	I2C1 event interrupt	0x00000078
31	17	programmable	I2C1_ER	I2C1 error interrupt	0x0000007C
32	18	programmable	USART1	USART1 global interrupt	0x00000080
33	19	programmable	EXTI15_8	EXTI line 8-15 interrupt	0x00000084
34	20	programmable	TIM1BRK	TIM1 brake interrupt	0x00000088
35	21	programmable	TIM1UP	TIM1 upgrade interrupt	0x0000008C
36	22	programmable	TIM1TRG	TIM1 trigger interrupt	0x00000090
37	23	programmable	TIM1CC	TIM1 capture/compare interrupt	0x00000094
38	24	programmable	TIM2	TIM2 global interrupt	0x00000098
39	25	programmable	USBPD	USBPD interrupt	0x0000009C
40	26	programmable	USBPD_WKUP	USBPD wakeup interrupt	0x000000A0

5.4 External Interrupt and Event Controller (EXTI)

5.4.1 Overview

Figure 5-1 External interrupt (EXTI) interface block diagram



As can be seen from Figure 5-1, the trigger source of the external interrupt can be either a software interrupt (SWIEVR) or an actual external interrupt channel. The signal of the external interrupt channel will be screened by the edge detect circuit first. Whenever one of the software interrupt or external interrupt signals is generated, it will be output to two AND gate circuits, event enable and interrupt enable, through the OR gate circuit in the figure, as

long as an interrupt is enabled or an event is enabled, an interrupt or an event will be generated. six registers of EXTI are accessed by the processor through the HB interface.

5.4.2 Wake-up Event

The system can wake up the Sleep mode caused by the WFE command through a wake-up event. The wake-up event is generated by either of the following two configurations.

- Enable an interrupt in a peripheral register, but not enabling this interrupt in the PFIC of the core, and enabling the SEVONPEND bit in the core at the same time. Embodied in EXTI, it is to enable an EXTI interrupt, but not to enable the EXTI interrupt in PFIC, and to enable the SEVONPEND bit at the same time. When the CPU wakes up from WFE, it needs to clear the EXTI interrupt flag bit and the PFIC pending bit.
- Enable an EXTI channel as an event channel eliminates the need for the CPU to clear the interrupt flag bit and the PFIC pending bit after waking up from the WFE.

5.4.3 Description

Using an external interrupt requires configuring the corresponding external interrupt channel, i.e. selecting the corresponding trigger edge and enabling the corresponding interrupt. When the set trigger edge appears on the external interrupt channel, an interrupt request will be generated and the corresponding interrupt flag bit will be set. The flag bit can be cleared by writing 1 to the flag bit.

Steps for using external hardware interrupts.

1) Configuration of GPIO operations.
2) Configure the interrupt enable bit (EXTI_INTENR) for the corresponding external interrupt channel.
3) Configuring the trigger edge (EXTI_RTENR or EXTI_FTENR) to select rising edge trigger, falling edge trigger or double edge trigger.
4) Configure EXTI interrupts in the core's PFIC to ensure they can respond correctly.

Steps for using external hardware events.

1) Configuration of GPIO operations.
2) Configure the event enable bit (EXTI_EVENR) for the corresponding external interrupt channel.
3) Configure the trigger edge (EXTI_RTENR or EXTI_FTENR) to select rising edge trigger, falling edge trigger, or double edge trigger.

Using the software interrupt/event steps.

1) Enabling external interrupts (EXTI_INTENR) or external events (EXTI_EVENR).
2) If using interrupt service functions, the EXTI interrupt needs to be set in the core's PFIC.
3) Set the software interrupt trigger (EXTI_SWIEVR), that is, an interrupt will be generated.

5.4.4 External Event Mapping

Table 5-2 EXTI Interrupt Mapping

External interrupt/Event lines	Mapping event description
EXTI0~EXTI15	Px0~Px15 (x=A/B), any of the I/O ports can be enabled for external interrupt/event functions, configured by the AFIO_EXTICRx register.
EXTI16	PVD event: voltage monitoring threshold exceeded
EXTI17	AWU auto-wakeup event
EXTI18	USB PD wakeup event

5.5 Register Description

5.5.1 EXTI Registers

Table 5-3 EXTI-related registers list

Name	Access address	Description	Reset value
R32_EXTI_INTENR	0x40010400	Interrupt enable register	0x00000000
R32_EXTI_EVENR	0x40010404	Event enable register	0x00000000
R32_EXTI_RTENR	0x40010408	Rising edge trigger enable register	0x00000000
R32_EXTI_FTENR	0x4001040C	Falling edge trigger enable register	0x00000000
R32_EXTI_SWIEVR	0x40010410	Soft interrupt event register	0x00000000
R32_EXTI_INTFR	0x40010414	Interrupt flag register	0x0000XXXX

5.5.1.1 Interrupt Enable Register (EXTI_INTENR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													MR18	MR17	MR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0

Bit	Name	Access	Description	Reset value
[31:19]	Reserved	RO	Reserved	0
[18:0]	MRx	RW	Enable the interrupt request signal for external interrupt channel x. 1: Enable interrupts for this channel. 0: Mask interrupts for this channel.	0

5.5.1.2 Event Enable Register (EXTI_EVENR)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													MR18	MR17	MR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0

Bit	Name	Access	Description	Reset value
[31:19]	Reserved	RO	Reserved.	0
[18:0]	MRx	RW	Enable the event request signal for external interrupt channel x. 1: Enable events for this channel. 0: Mask events for this channel.	0

5.5.1.3 Rising Edge Trigger Enable Register (EXTI_RTENR)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													TR18	TR17	TR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0

Bit	Name	Access	Description	Reset value
[31:19]	Reserved	RO	Reserved.	0
[18:0]	TRx	RW	Enable rising edge triggering for external interrupt channel x. 1: Enable rising edge triggering for this channel. 0: Disable rising edge triggering for this channel.	0

5.5.1.4 Falling Edge Trigger Enable Register (EXTI_FTENR)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													TR18	TR17	TR16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0

Bit	Name	Access	Description	Reset value
[31:19]	Reserved	RO	Reserved	0
[18:0]	TRx	RW	Enable falling edge triggering for external interrupt channel x. 0: Disable falling edge triggering for this channel. 1: Enable falling edge triggering for this channel.	0

5.5.1.5 Software Interrupt Event Register (EXTI_SWIEVR)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved													SWIER18	SWIER17	SWIER16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWIER15	SWIER14	SWIER13	SWIER12	SWIER11	SWIER10	SWIER 9	SWIER 8	SWIER 7	SWIER 6	SWIER 5	SWIER 4	SWIER 3	SWIER 2	SWIER 1	SWIER 0

Bit	Name	Access	Description	Reset value
[31:19]	Reserved	RO	Reserved	0
[18:0]	SWIERx	RW	A software interrupt is set on the corresponding externally triggered interrupt channel. Setting it here causes the interrupt flag bit (EXTI_INTFR) to correspond to the position bit, and if interrupt enable (EXTI_INTENR) or event enable (EXTI_EVENR) is on, then an interrupt or event will be generated.	0

5.5.1.6 Interrupt Flag Register (EXTI_INTFR)

Offset address: 0x14

Bit	Name	Access	Description	Reset value
[31:19]	Reserved	RO	Reserved	0
[18:0]	IFx	W1	The interrupt flag bit, this location bit flags that a corresponding external interrupt has occurred. A write of 1 clears this bit.	X

5.5.2 PFIC Registers

Table 5-4 PFIC-related registers list

Name	Access address	Description	Reset value
R32_PFIC_ISR1	0xE000E000	PFIC interrupt enable status register 1	0x0000000C
R32_PFIC_ISR2	0xE000E004	PFIC interrupt enable status register 2	0x00000000
R32_PFIC_IPR1	0xE000E020	PFIC interrupt pending status register 1	0x00000000
R32_PFIC_IPR2	0xE000E024	PFIC interrupt pending status register 2	0x00000000
R32_PFIC_ITHRESDR	0xE000E040	PFIC interrupt priority threshold configuration register	0x00000000
R32_PFIC_CFGR	0xE000E048	PFIC interrupt configuration register	0x00000000
R32_PFIC_GISR	0xE000E04C	PFIC interrupt global status register	0x00000000
R32_PFIC_VTFIDR	0xE000E050	PFIC VTF interrupt ID configuration register	0x00000000
R32_PFIC_VTFADDRR0	0xE000E060	PFIC VTF interrupt 0 offset address register	0x00000000
R32_PFIC_VTFADDRR1	0xE000E064	PFIC VTF interrupt 1 offset address register	0x00000000
R32_PFIC_IENR1	0xE000E100	PFIC interrupt enable setting register 1	0x00000000
R32_PFIC_IENR2	0xE000E104	PFIC interrupt enable setting register 2	0x00000000
R32_PFIC_IRER1	0xE000E180	PFIC interrupt enable clear register 1	0x00000000
R32_PFIC_IRER2	0xE000E184	PFIC interrupt enable clear register 2	0x00000000
R32_PFIC_IPSR1	0xE000E200	PFIC interrupt pending setting register 1	0x00000000
R32_PFIC_IPSR2	0xE000E204	PFIC interrupt pending setting register 2	0x00000000
R32_PFIC_IPRR1	0xE000E280	PFIC interrupt pending clear register 1	0x00000000
R32_PFIC_IPRR2	0xE000E284	PFIC interrupt pending clear register 2	0x00000000
R32_PFIC_IACTR1	0xE000E300	PFIC interrupt activation status register 1	0x00000000
R32_PFIC_IACTR2	0xE000E304	PFIC interrupt activation status register 2	0x00000000
R32_PFIC_IPRIORx	0xE000E400	PFIC interrupt priority configuration register	0x00000000
R32_PFIC_SCTLR	0xE000ED10	PFIC system control register	0x00000000

Note: 1. 1. The default value of the PFIC_ISR1 register is 0xC, that is, exceptions are always enabled by default.

2. EXC supports interrupt pending clear and set operations, but does not support interrupt enable clear and set operations.

5.5.2.1 PFIC Interrupt Enable Status Register 1 (PFIC_ISR1)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INTENSTA[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INTENSTA14	Reserved	INTENSTA12	Reserved								INTENSTA3	INTENSTA2	Reserved	

Bit	Name	Access	Description	Reset value
[31:16]	INTENSTA	RO	16#-31# interrupt current enable status.	0

			1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	
15	Reserved	RO	Reserved	0
14	INTENSTA	RO	14# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0
13	Reserved	RO	Reserved	0
12	INTENSTA	RO	12# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0
[11:4]	Reserved	RO	Reserved	0
[3:2]	INTENSTA	RO	2#-3# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0
[1:0]	Reserved	RO	Reserved	0

5.5.2.2 PFIC Interrupt Enable Status Register 2 (PFIC_ISR2)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									INTENSTA[6:0]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	INTENSTA	RO	32#-38# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0

5.5.2.3 PFIC Interrupt Pending Status Register 1 (PFIC_IPR1)

Offset address: 0x20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PENDSTA[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	PENDSTA14	Reserved	PENDSTA12	Reserved								PENDSTA3	PENDSTA2	Reserved	

Bit	Name	Access	Description	Reset value
[31:16]	PENDSTA	RO	16#-31# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0
15	Reserved	RO	Reserved	0
14	PENDSTA	RO	14# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0
13	Reserved	RO	Reserved	0
12	PENDSTA	RO	12# interrupt current enable status. 1: Current numbered interrupt is enabled.	0

			0: Current numbered interrupt is not enabled.	
[11:4]	Reserved	RO	Reserved	0
[3:2]	PENDSTA	RO	2#-3# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0
[1:0]	Reserved	RO	Reserved	0

5.5.2.4 PFIC Interrupt Pending Status Register 2 (PFIC_IPR2)

Offset address: 0x24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									PENDSTA[38:32]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	PENDSTA	RO	32#-38# interrupt current enable status. 1: Current numbered interrupt is enabled. 0: Current numbered interrupt is not enabled.	0

5.5.2.5 PFIC Interrupt Priority Threshold Configuration Register (PFIC_ITHRESDR)

Offset address: 0x40

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved																								THRESHOLD[7:0]							

Bit	Name	Access	Description	Reset value
[31:8]	Reserved	RO	Reserved	0
[7:0]	THRESHOLD[7:0]	RW	Interrupt priority threshold setting value. The interrupt priority value lower than the current setting value, when hung, does not perform interrupt service; this register is 0 means the threshold register function is invalid. [7:6]: Priority threshold; [5:0]: Reserved, fixed to 0, write invalid.	0

5.5.2.6 PFIC Interrupt Configuration Register (PFIC_CFGR)

Offset address: 0x48

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYCODE[15:0]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								RSTSYS	Reserved						

Bit	Name	Access	Description	Reset value
[31:16]	KEYCODE[15:0]	WO	Corresponding to different target control bits, the	0

			corresponding security access identification data needs to be written simultaneously in order to be modified, and the readout data is fixed to 0. KEY1 = 0xFA05; KEY2 = 0xBCAF; KEY3 = 0xBEEF.	
[15:8]	Reserved	RO	Reserved.	0
7	RSTSYS	WO	System reset (simultaneous writing to KEY3). Auto clear 0. Writing 1 is valid, writing 0 is invalid. *Note: Same function as the PFIC_SCTLR register SYSRST bit.*	0
[6:0]	Reserved	RO	Reserved.	0

5.5.2.7 PFIC Interrupt Global Status Register (PFIC_GISR)

Offset address: 0x4C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						GPEND STA	GACT STA	NESTSTA[7:0]							

Bit	Name	Access	Description	Reset value
[31:10]	Reserved	RO	Reserved	0
9	GPENDSTA	RO	Are there any interrupts currently on hold. 1: Yes; 0: No.	0
8	GACTSTA	RO	Are there any interrupts currently being executed. 1: Yes; 0: No.	0
[7:0]	NESTSTA[7:0]	RO	Current interrupt nesting status, currently supports a maximum of 2 levels of nesting and a maximum hardware stack depth of 2 levels. 0x03: Level 2 interrupt in progress. 0x01: Level 1 interrupt in progress. Other: Unlikely cases.	0

5.5.2.8 PFIC VTF Interrupt ID Configuration Register (PFIC_VTFIDR)

Offset address: 0x50

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VTFID1								VTFID0							

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:8]	VTFID1	RW	Configure the interrupt number of VTF interrupt 1.	0
[7:0]	VTFID0	RW	Configure the interrupt number of VTF	0

			interrupt 0.	

5.5.2.9 PFIC VTF Interrupt 0 Address Register (PFIC_VTFADDRR0)

Offset address: 0x60

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR0[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR0[15:1]															VTF0EN

Bit	Name	Access	Description	Reset value
[31:1]	ADDR0[31:1]	RW	VTF interrupt 0 service program address bit[31:1], bit0 is 0.	0
0	VTF0EN	RW	VTF interrupt 0 enable bit. 1: Enable VTF interrupt 0 channel; 0: off.	0

5.5.2.10 PFIC VTF Interrupt 1 Address Register (PFIC_VTFADDRR1)

Offset address: 0x64

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR1[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR1[15:1]															VTF1EN

Bit	Name	Access	Description	Reset value
[31:1]	ADDR1[31:1]	RW	VTF interrupt 1 service program address bit[31:1], bit0 is 0.	0
0	VTF1EN	RW	VTF interrupt 1 enable bit. 1: VTF interrupt 1 channel is enabled; 0: Off.	0

5.5.2.11 PFIC Interrupt Enable Setting Register 1 (PFIC_IENR1)

Offset address: 0x100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INTEN[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INTEN14	Reserved	INTEN12	Reserved											

Bit	Name	Access	Description	Reset value
[31:16]	INTEN	WO	16#-31# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0
15	Reserved	RO	Reserved	0

14	INTEN	WO	14# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0
13	Reserved	RO	Reserved	0
12	INTEN	WO	12# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0
[11:0]	Reserved	RO	Reserved	0

5.5.2.12 PFIC Interrupt Enable Setting Register 2 (PFIC_IENR2)

Offset address: 0x104

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									INTEN[38:32]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	INTEN	WO	32#-38# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0

5.5.2.13 PFIC Interrupt Enable Clear Register 1 (PFIC_IRER1)

Offset address: 0x180

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INTRSET[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	INTRSET14	Reserved	INTRSET12	Reserved											

Bit	Name	Access	Description	Reset value
[31:16]	INTRSET	WO	16#-31# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0
15	Reserved	RO	Reserved	0
14	INTRSET	WO	14# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0
13	Reserved	RO	Reserved	0
12	INTRSET	WO	12# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0
[11:0]	Reserved	RO	Reserved	0

5.5.2.14 PFIC Interrupt Enable Clear Register 2 (PFIC_IRER2)

Offset address: 0x184

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									INTRSET[38:32]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	INTRSET	WO	32#-38# interrupt enable control. 1: Current number interrupt enable. 0: No effect.	0

5.5.2.15 PFIC Interrupt Pending Setting Register 1 (PFIC_IPSR1)

Offset address: 0x200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PENDSET[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	PENDSET14	Reserved	PENDSET12	Reserved								PENDSET3	PENDSET2	Reserved	

Bit	Name	Access	Description	Reset value
[31:16]	PENDSET	WO	16#-31# interrupt pending settings: 1: Current number interrupt pending; 0: No effect.	0
15	Reserved	RO	Reserved	0
14	PENDSET	WO	14# interrupt pending settings: 1: Current number interrupt pending; 0: No effect.	0
13	Reserved	RO	Reserved	0
12	PENDSET	WO	12# interrupt pending settings: 1: Current number interrupt pending; 0: No effect.	0
[11:4]	Reserved	RO	Reserved	0
[3:2]	PENDSET	WO	2#-3# interrupt pending settings: 1: Current number interrupt pending; 0: No effect.	0
[1:0]	Reserved	RO	Reserved	0

5.5.2.16 PFIC Interrupt Pending Setting Register 2 (PFIC_IPSR2)

Offset address: 0x204

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									PENDSET[38:32]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	PENDSET	WO	32#-38# interrupt pending settings:	0

			1: Current number interrupt pending; 0: No effect.	

5.5.2.17 PFIC Interrupt Pending Clear Register 1 (PFIC_IPRR1)

Offset address: 0x280

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PENDRST[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	PENDRST14	Reserved	PENDRST12	Reserved								PENDRST3	PENDRST2	Reserved	

Bit	Name	Access	Description	Reset value
[31:16]	PENDRST	WO	16#-31# interrupt pending cleared: 1: Current numbered interrupt clears the pending status; 0: No effect.	0
15	Reserved	RO	Reserved	0
14	PENDRST	WO	14# interrupt pending cleared: 1: Current numbered interrupt clears the pending status; 0: No effect.	0
13	Reserved	RO	Reserved	0
12	PENDRST	WO	12# interrupt pending cleared: 1: Current numbered interrupt clears the pending status; 0: No effect.	0
[11:4]	Reserved	RO	Reserved	0
[3:2]	PENDRST	WO	2#-3# interrupt pending cleared: 1: Current numbered interrupt clears the pending status; 0: No effect.	0
[1:0]	Reserved	RO	Reserved	0

5.5.2.18 PFIC Interrupt Pending Clear Register 2 (PFIC_IPRR2)

Offset address: 0x284

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									PENDRST[38:32]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	PENDRST	WO	32#-38# interrupt pending cleared: 1: Current numbered interrupt clears the pending status; 0: No effect.	0

5.5.2.19 PFIC Interrupt Activation Status Register 1 (PFIC_IACTR1)

Offset address: 0x300

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IACTS [31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	IACTS14	Reserved	IACTS12	Reserved								IACTS3	IACTS2	Reserved	

Bit	Name	Access	Description	Reset value
[31:16]	IACTS	RO	16#-31# interrupt execution status. 1: Current number interrupt in execution. 0: Current number interrupt is not executed.	0
15	Reserved	RO	Reserved	0
14	IACTS	RO	14# interrupt execution status. 1: Current number interrupt in execution. 0: Current number interrupt is not executed.	0
13	Reserved	RO	Reserved	0
12	IACTS	RO	12# interrupt execution status. 1: Current number interrupt in execution. 0: Current number interrupt is not executed.	0
[11:4]	Reserved	RO	Reserved	0
[3:2]	IACTS	RO	2#-3# interrupt execution status. 1: Current number interrupt in execution. 0: Current number interrupt is not executed.	0
[1:0]	Reserved	RO	Reserved	0

5.5.2.20 PFIC Interrupt Activation Status Register 2 (PFIC_IACTR2)

Offset address: 0x304

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									IACTS [38:32]						

Bit	Name	Access	Description	Reset value
[31:7]	Reserved	RO	Reserved	0
[6:0]	IACTS	RO	32#-38# interrupt execution status. 1: Current number interrupt in execution. 0: Current number interrupt is not executed.	0

5.5.2.21 PFIC Interrupt Priority Configuration Register (PFIC_IPRIORx) (x=0-63)

Offset address: 0x400-0x4FF

The controller supports 256 interrupts (0-255), each using 8 bits to set the control priority.

	31 24	23 16	15 8	7 0
IPRIOR63	PRIO_255	PRIO_254	PRIO_253	PRIO_252
…	…	…	…	…
IPRIORx	PRIO_(4x+3)	PRIO_(4x+2)	PRIO_(4x+1)	PRIO_(4x)

…	…	…	…	…
IPRIOR0	PRIO_3	PRIO_2	PRIO_1	PRIO_0

Bit	Name	Access	Description	Reset value
[2047:2040]	IP_255	RW	Same as IP_0 description.	0
…	…	…	…	…
[31:24]	IP_3	RW	Same as IP_0 description.	0
[23:16]	IP_2	RW	Same as IP_0 description.	0
[15:8]	IP_1	RW	Same as IP_0 description.	0
[7:0]	IP_0	RW	Number 0 interrupt priority configuration. [7:6]: Priority control bits. If no nesting is configured, no preemption bits; If configured with 2 levels of nesting, bit 7 is the preemption bit; [5:0] : Reserved, fixed to 0, write invalid.	0

5.5.2.22 PFIC System Control Register (PFIC_SCTLR)

Offset address: 0xD10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SYSRST	Reserved														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										SETEVENT	SEVONPEND	WFITOWFE	SLEEPDEEP	SLEEPONEXIT	Reserved

Bit	Name	Access	Description	Reset value
31	SYSRST	WO	System reset, clear 0 automatically. write 1 valid, write 0 invalid, same effect as PFIC_CFGR register.	0
[30:6]	Reserved	RO	Reserved.	0
5	SETEVENT	WO	Set the event to wake up the WFE case.	0
4	SEVONPEND	RW	When an event occurs or interrupts a pending state, the system can be woken up from after the WFE instruction, or if the WFE instruction is not executed, the system will be woken up immediately after the next execution of the instruction. 1: enabled events and all interrupts (including unenabled interrupts) can wake up the system. 0: Only enabled events and enabled interrupts can wake up the system.	0
3	WFITOWFE	RW	Execute the WFI command as if it were a WFE. 1: treat the subsequent WFI instruction as a WFE instruction. 0: No effect.	0
2	SLEEPDEEP	RW	Low-power mode of the control system. 1：deepsleep 0：sleep	0
1	SLEEPONEXIT	RW	System status after control leaves the interrupt	0

			service program. 1: The system enters low-power mode. 0: The system enters the main program.	
0	Reserved	RO	Reserved.	0

5.5.3 Dedicated CSR Registers

A number of Control and Status Registers (CSRs) are defined in the RISC-V architecture to configure or identify or record operational status. In addition to the standard registers defined in the RISC-V Privileged Architecture document, the CH641 chip also has a number of vendor-defined registers that need to be accessed using the csr instruction.

Note: These registers are labeled "MRW, MRO, MRW1" and require the system to be in machine mode to access them.

5.5.3.1 Interrupt System Control Register (INTSYSCR)

CSR address: 0x804

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														INESTEN	HWSTKEN

Bit	Name	Access	Description	Reset value
[31:2]	Reserved	MRO	Reserved	0
1	INESTEN	MRW	Interrupt nesting enable: 0: Interrupt nesting function disabled; 1: Interrupt nesting function enabled.	0
0	HWSTKEN	MRW	Hardware stack enable: 0: Hardware stacking function disabled; 1: Hardware stacking function enabled.	0

5.5.3.2 Exception Entry Base Address Register (MTVEC)

CSR address: 0x305

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BASEADDR[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BASEADDR[15:2]														MODE1	MODE0

Bit	Name	Access	Description	Reset value
[31:2]	BASEADDR[31:2]	MRW	Interrupt vector table base address.	0
1	MODE1	MRW	Interrupt vector table identifies patterns. 0: Identify by jump instruction, limited range, support for non-jump instructions. 1: Identify by absolute address, support full range, but must jump.	0
0	MODE0	MRW	Interrupt or exception entry address mode	0

			selection. 0: Use of a unified entry address. 1: Address offset based on interrupt number *4.	

5.5.4 STK Register Description

Table 5-5 STK-related registers list

Name	Access address	Description	Reset value
R32_STK_CTLR	0xE000F000	System count control register	0x00000000
R32_STK_SR	0xE000F004	System count status register	0x00000000
R32_STK_CNTL	0xE000F008	System counter register	0x00000000
R32_STK_CMPLR	0xE000F010	Count/compare register	0x00000000

5.5.4.1 System Count Control Register (STK_CTLR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SWIE	Reserved														

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												STRE	STCLK	STIE	STE

Bit	Name	Access	Description	Reset value
31	SWIE	RW	Software interrupt trigger enable (SWI). 1: Triggering software interrupts. 0: Turn off the trigger. After entering software interrupt, software clear 0 is required, otherwise it is continuously triggered.	0
[30:4]	Reserved	RO	Reserved.	
3	STRE	RW	Auto-reload count enable bit. 1: Re-counting from 0 after counting up to the comparison value. 0: Count up to the comparison value and continue counting up, count down to 0 and start counting down again from the maximum value.	
2	STCLK	RW	Counter clock source selection bit. 1: HCLK for time base. 0: HCLK/8 for time base.	
1	STIE	RW	Counter interrupt enable control bit. 1: Enable counter interrupt. 0: Disable counter interrupt.	
0	STE	RW	System counter enable control bit. 1: Turn on the system counter STK. 0: Turn off the system counter STK and the counter stops counting.	0

5.5.4.2 System Count Status Register (STK_SR)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															CNTIF

Bit	Name	Access	Description	Reset value
[31:1]	Reserved	RO	Reserved	0
0	CNTIF	RW0	Count value comparison flag, write 0 to clear, write 1 to invalidate. 1: Counting up to the comparison value and down to 0; 0: The comparison value is not reached.	0

5.5.4.3 System Counter Register (STK_CNTL)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNT[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	CNT[31:0]	RW	The current counter count value is 32 bits.	0

5.5.4.4 Count/Compare Register (STK_CMPLR)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMP[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	CMP[31:0]	RW	Set compare counter value to 32-bit.	0

Chapter 6 GPIO and Alternate Function (GPIO/AFIO)

The GPIO port can be configured into a variety of input or output modes, with a built-in pull-up or pull-down resistor that can be turned off, and can be configured as a push-pull output. The GPIO port can also be reused for other functions.

PB2H, PB3H, PB4H, and PB5H are V_{HV}-powered high-voltage I/O pins, and the rest are V_{DD}-powered low-voltage I/O pins.

PA2, PA3, PA4, PA5, and PA9 all have strong current driving capabilities, which is about twice that of other ordinary I/O pins.

When PB8 is used as a GPIO, it only supports input or open-drain output, and does not support push-pull output.

PB6 has no GPPB3H and PB4H have built-in pull-up resistors that cannot be turned off; PB2H and PB5H do not have built-in pull-up resistors; ISP and PB8/QII do not have built-in pull-up resistors; PA0 and PA1 have built-in pull-up resistors that are turned off by default and can be adjusted, and are controlled by 2 EXTEN_CTLR1 PUE and DAC are adjusted and controlled, and can provide pull-up current; 3 CC pins are provided, of which PB0/CC1R and PB9/CC3R pins have built-in controllable Rd pull-down resistors defined by the type-C specification, and PB1/CC2 pins The pin does not provide Rd by default, but can support customization and is controlled by CC_PU in R8_PORT_CC of the corresponding pin; other GPIO pins have built-in pull-up resistors that are turned off by default and can be turned on.IO function and is used only for ISP input.

PA4 and PA5 have built-in pull-down resistors that are enabled by default and can be disabled; PA0 and PA1 have built-in pull-down resistors that are enabled by default and can be adjusted or disabled, which are adjusted and controlled by two sets of PDE and DAC in EXTEN_CTLR1 and can provide pull-down current; PB0/CC1R and PB9/CC3R pins have built-in controllable Rd pull-down resistors defined by the default type-C specification, which are enabled by default and need to be disabled when used as GPIO push-pull outputs. PB1/CC2 pins do not provide Rd by default, but can be customized and controlled by CC_PD in R8_PORT_CC of the corresponding pins; other GPIO pins do not have built-in pull-down resistors.

6.1 Main Features

Each pin of the port can be configured to one of the following multiple modes.

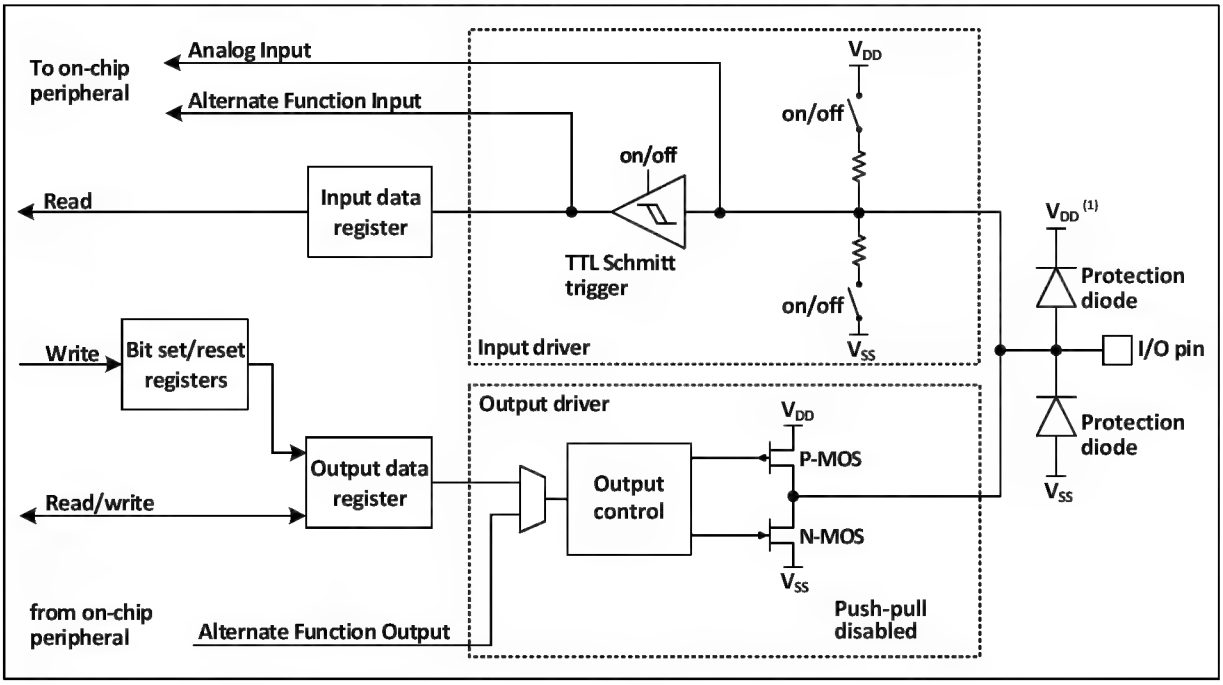
- Floating input
- Pull-up input
- Pull-down input (partial IO)
- Analog input
- Push-pull output
- Inputs and outputs of alternate function

Many pins have alternate capabilities, and many other peripherals map their output and input channels to these pins. The specific usage of these alternate pins needs to be referred to the individual peripherals, and the content of whether these pins are alternate and remapped is explained in this chapter.

6.2 Function Description

6.2.1 Overview

Figure 6-1 GPIO module basic structure block diagram



Note: (1) V_{DD} is V_{DD} when GPIO is normal IO, and V_{DD} is V_{HV} when GPIO is HV high voltage IO.

As shown in Figure 6-1, the IO port structure is shown. Each pin has two protection diodes inside the chip. The IO port can be divided into input and output driver modules. The input driver has optional weak pull-up and pull-down resistors, which can be connected to analog input peripherals such as ADC; if the input is to a digital peripheral, it needs to go through a TTL Schmitt trigger and then be connected to a GPIO input register or other multiplexing peripherals. The output driver has a pair of MOS tubes, and the IO port can be configured as a push-pull output by configuring the upper and lower MOS tubes; the output driver can also be configured internally to be controlled by GPIO or by other multiplexed peripherals.

6.2.2 GPIO Initialization Function

Just after reset, the GPIO port runs in the initial state. At this time, most IO ports run in the floating input state. For specific initialization functions, please refer to the relevant chapters of the pin description.

6.2.3 External Interrupts

All GPIO ports can be configured with external interrupt input channels, but an external interrupt input channel can only be mapped to at most one GPIO pin, and the serial number of the external interrupt channel must be consistent with the bit number of the GPIO port, such as PA1 (or PB1) can only be mapped to EXTI1, and EXTI1 can only accept the mapping of one of PA1 and PB1. Both parties have a one-to-one relationship.

6.2.4 Alternate Functions

It is important to note that using the alternate function.

- To use the alternate function in the input direction, the port must be configured in alternate input mode, and

the pull-down settings can be set according to actual needs.

- To use the alternate function in the output direction, the port must be configured in alternate output mode.
- For bidirectional alternate function, the port must be configured in alternate output mode, when the driver is configured in floating input mode

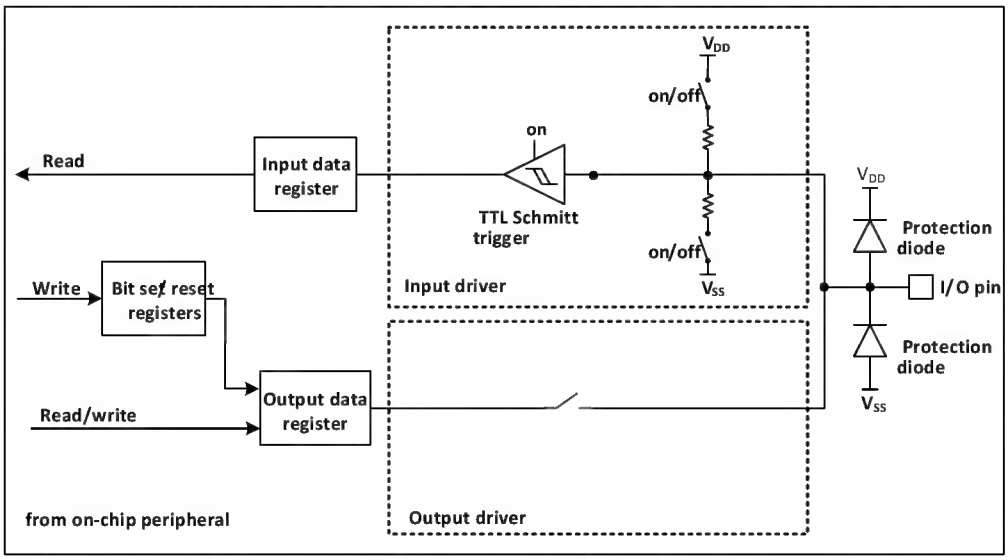
The same I/O port may have multiple peripherals alternate to this pin, so in order to maximize the space for each peripheral, the alternate pins of peripherals can be remapped to other pins in addition to the default alternate pins, avoiding the occupied pins.

6.2.5 Locking Mechanism

The locking mechanism locks the configuration of the I/O port. After a specific write sequence, the selected I/O pin configuration will be locked and cannot be changed until the next reset.

6.2.6 Input Configuration

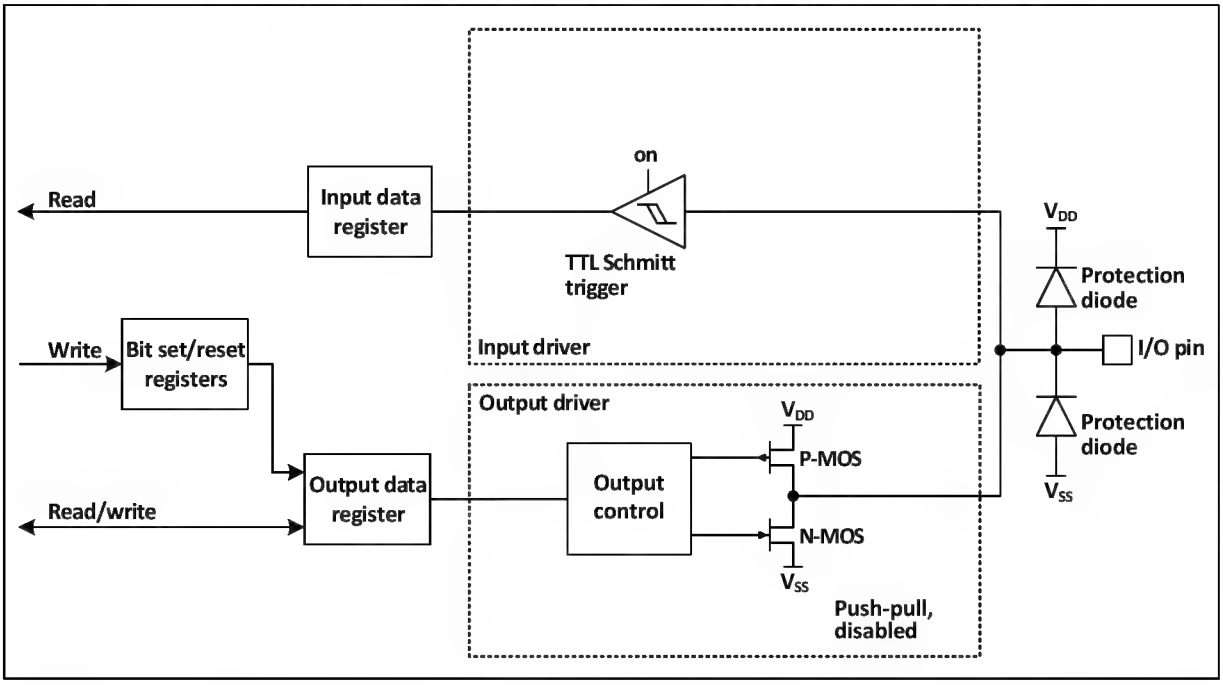
Figure 6-2 GPIO module input configuration structure block diagram



When the IO port is configured in input mode, the output driver is disconnected, the input pull-up and pull-down are selectable, and no alternate functions or analogue inputs are connected. The data on each IO port is sampled at each HB clock into the input data register and the level status of the corresponding pin is obtained by reading the corresponding bit in the input data register.

6.2.7 Output Configuration

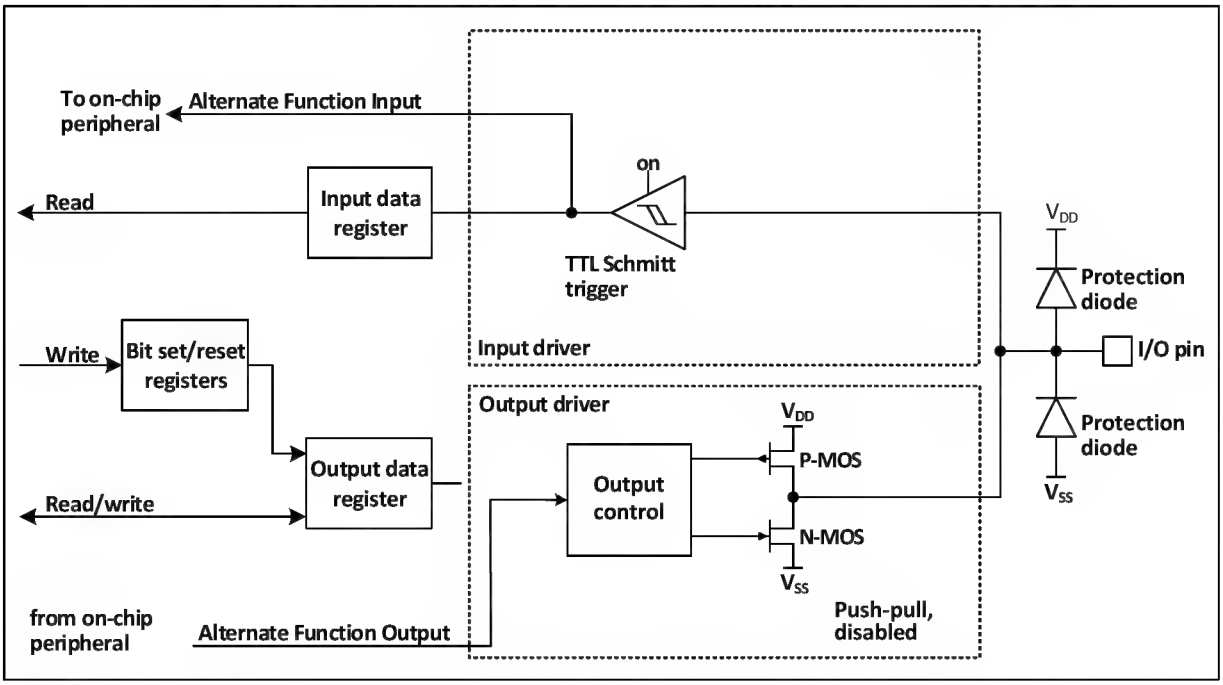
Figure 6-3 GPIO module output configuration structure block diagram



When the IO port is configured in output mode, a pair of MOS in the output driver is configured in push-pull mode, and the alternate function is not used. The input driver's pull-up and pull-down resistors are disabled, the TTL Schmitt trigger is activated and the levels appearing on the IO pins will be sampled into the input data registers at each HB clock, so reading the input data registers will give the IO status and access to the output data registers will give the last written value.

6.2.8 Alternate Function Configuration

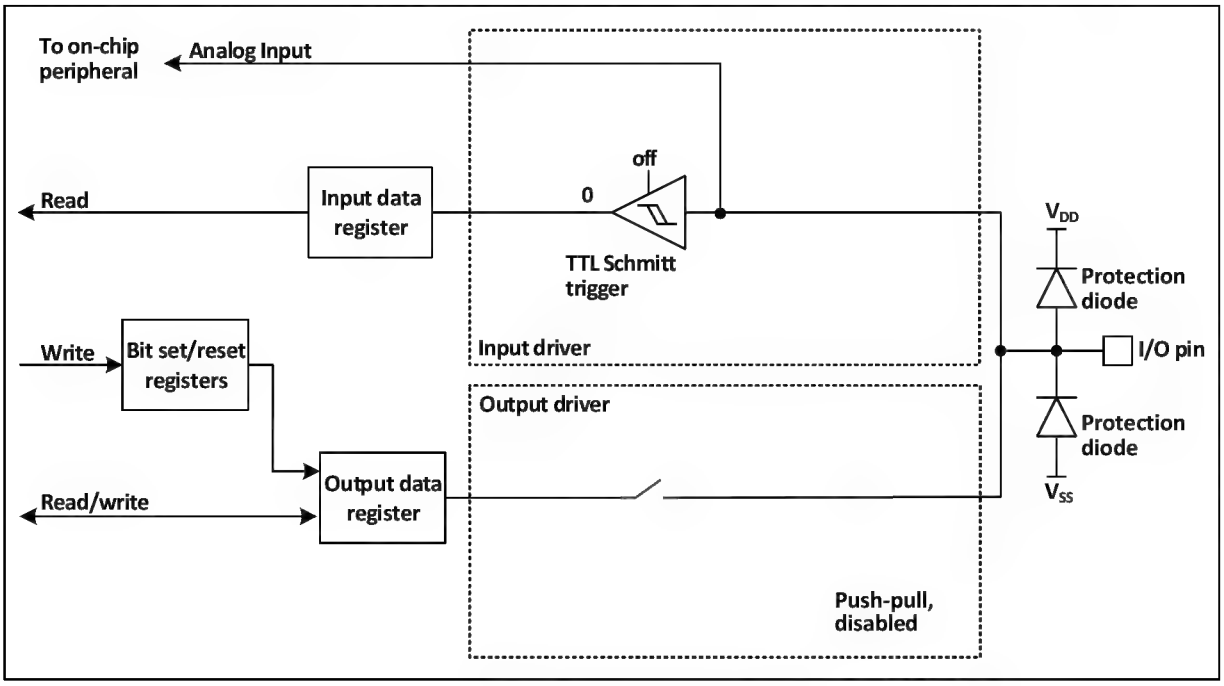
Figure 6-4 The structure of GPIO module when it is alternate by other peripherals



When alternate is enabled, the output drivers are enabled, configured in push-pull mode, or automatically configured in open-drain mode if used for I2C, the Schmitt trigger is turned on, the input and output lines of the alternate function are connected, but the output data registers are disconnected, the levels present on the IO pins will be sampled into the input data registers at each HB clock, and reading the input data registers will give the current status of the IO port; in push-pull mode, reading the output data registers will give the last written value.

6.2.9 Analog Input Configuration

Figure 6-5 The configuration structure when the GPIO module is used as an analog input



6.2.10 GPIO Settings for Peripherals

The following table recommends the corresponding GPIO port configuration for each peripheral pin.

Table 6-1 Advanced-control timer (TIM1)

TIM1	Configuration	GPIO configuration
TIM1_CHx	Input capture channel x	Floating input
	Output compare channel x	Push-pull alternate output
TIM1_CHxN	Complementary output channels x	Push-pull alternate output
TIM1_BKIN	Brake input	Floating input
TIM1_ETR	Externally triggered clock input	Floating input

Table 6-2 General-purpose timer (TIM2)

TIM2 pin	Configuration	GPIO configuration
TIM2_CHx	Input capture channel x	Floating input
	Output compare channel x	Push-pull alternate output

Table 6-3 Universal synchronous asynchronous serial transceiver (USART)

USART1 pin	Configuration	GPIO configuration

USART1_TX	Full-duplex mode	Push-pull alternate output
	Half-duplex synchronous mode	Alternate push-pull output (module realizes open-drain output)
USART1_RX	Full-duplex mode	Floating input or pull-up input
	Half-duplex synchronous mode	Not used
USART1_RTS	Hardware flow control	Push-pull alternate output
USART1_CTS	Hardware flow control	Floating input or pull-up input

Table 6-4 Internal integrated bus (I2C1) module

I2C1 pin	Configuration	GPIO configuration
I2C1_SCL	I2C1 clock	Alternate push-pull output (module realizes open-drain output)
I2C1_SDA	I2C1 data	

Table 6-5 Analog-to-digital converters (ADC)

ADC pin	GPIO configuration
ADC	Analog input

Table 6-6 Other I/O function settings

pin	Configuration	GPIO configuration
MCO	Clock output	Push-pull alternate output
EXTI	External interrupt input	Float, pull-up or pull-down input

6.2.11 Alternate Function Remapping GPIO Settings

6.2.11.1 Timer Alternate Function Remapping

Table 6-7 TIM1 alternate function remapping

Alternate function	TIM1_RM=0 Default mapping	TIM1_RM=1 Remapping
TIM1_ETR	PB9	
TIM1_BKIN	PA15	
TIM1_CH1	PB2(HV high voltage drive)	
TIM1_CH2	PB3(HV high voltage drive)	
TIM1_CH3	PB4(HV high voltage drive)	PA2(LV low voltage strong drive)
TIM1_CH1N	PA2(LV low voltage strong drive)	PB4(HV high voltage drive)
TIM1_CH2N	PA3(LV low voltage strong drive)	PB5(HV high voltage drive)
TIM1_CH3N	PA4(LV low voltage drive)	

Table 6-8 TIM2 alternate function remapping

Alternate function	TIM2_RM=00 Default mapping	TIM2_RM=01 Partial mapping	TIM2_RM=10 Partial mapping	TIM2_RM=11 Full mapping
TIM2_CH1	PA5 (LV low voltage strong drive)			
TIM2_CH2	PA9 (LV low voltage strong drive)	PA4 (LV low voltage strong drive)	PA9 (LV low voltage strong drive)	PA4 (LV low voltage strong

				drive)
TIM2_CH1N	PA7		PB5(HV high voltage drive)	
TIM2_CH2N	PA8		PB4(HV high voltage drive)	

6.2.11.2 USART Alternate Function Remapping

Table 6-9 USART1 alternate function remapping

Alternate function	USART1_RM=0 00 Default mapping	USART1_RM=0 01 Remapping	USART1_RM=0 10 Remapping	USART1_RM=0 11 Remapping	USART1_RM=1 00 Remapping
USART1_TX	PA11	PB1	PA0	PA1	PA2
USART1_RX	PA12	PB0	PA1	PA0	PA3
USART1_CTS	PA13				
USART1_RTS	PA14				

6.2.11.3 I2C1 Alternate Function Remapping

Table 6-10 I2C1 alternate function remapping

Alternate function	I2C1_RM=00 Default mapping	I2C1_RM=01 Remapping	I2C1_RM=1x Remapping
I2C1_SCL	PA13	PA2	PA11
I2C1_SDA	PA14	PA3	PA12

6.2.11.4 ADC Alternate Function Remapping

Table 6-11 ADC external trigger injection conversion alternate function remapping

Alternate function	ADC_RM=0 Default mapping	ADC_RM=1 Remapping
ADC external trigger injection conversion	ADC external trigger connected to PA4	ADC external trigger connected to PA15

6.3 Register Description

6.3.1 GPIO Register Description

Unless otherwise specified, the registers of the GPIO must be operated as words (operate these registers with 32 bits).

Table 6-12 GPIO-related registers list

Name	Access address	Description	Reset value
R32_GPIOA_CFGLR	0x40010800	PA port configuration register low	0x44444444
R32_GPIOA_CFGHR	0x40010804	PA port configuration register high	0x44884444
R32_GPIOB_CFGLR	0x40010C00	PB port configuration register low	0x44444444
R32_GPIOB_CFGHR	0x40010C04	PB port configuration register high	0x00000044
R32_GPIOA_INDR	0x40010808	PA port input data register	0x0000XXXX
R32_GPIOB_INDR	0x40010C08	PB port input data register	0x0000XXXX

R32_GPIOA_OUTDR	0x4001080C	PA port output data register	0x00000000
R32_GPIOB_OUTDR	0x40010C0C	PB port output data register	0x00000000
R32_GPIOA_BSHR	0x40010810	PA port set/reset register	0x00000000
R32_GPIOB_BSHR	0x40010C10	PB port set/reset register	0x00000000
R32_GPIOA_BCR	0x40010814	PA port reset register	0x00000000
R32_GPIOB_BCR	0x40010C14	PB port reset register	0x00000000
R32_GPIOA_LCKR	0x40010818	PA port configuration lock register	0x00000000
R32_GPIOB_LCKR	0x40010C18	PB port configuration lock register	0x00000000

6.3.1.1 GPIO Configuration Register Low (GPIOx_CFGLR) (x=A/B)

Offset address: 0x00

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
CNF7[1:0]	MODE7[1:0]	CNF6[1:0]	MODE6[1:0]	CNF5[1:0]	MODE5[1:0]	CNF4[1:0]	MODE4[1:0]

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
CNF3[1:0]	MODE3[1:0]	CNF2[1:0]	MODE2[1:0]	CNF1[1:0]	MODE1[1:0]	CNF0[1:0]	MODE0[1:0]

Bit	Name	Access	Description	Reset value
[31:30] [27:26] [23:22] [19:18] [15:14] [11:10] [7:6] [3:2]	CNFy[1:0]	RW	(y=0-7), the configuration bits for port x, by which the corresponding port is configured. When in input mode (MODE=00b). 00: Analog input mode. 01: Floating input mode. 10: With pull-up and pull-down mode. 11: Reserved. In output mode (MODE>00b). 0x: Universal push-pull output mode. 1x: Alternate function push-pull output mode, I2C1 automatically open drain.	The default value of CNF4 and CNF5 of GPIOA is 10b, and the other values are 01b.
[29:28] [25:24] [21:20] [17:16] [13:12] [9:8] [5:4] [1:0]	MODEy[1:0]	RW	(y=0-7), port x mode selection, configure the corresponding port through these bits. 00: Input mode; 01/10/11: Output mode, maximum speed 30MHz.	00b

6.3.1.2 GPIO Configuration Register High (GPIOx_CFGHR) (x=A/B)

Offset address: 0x04

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
CNF15[1:0]	MODE15 [1:0]	CNF14[1:0]	MODE14 [1:0]	CNF13[1:0]	MODE13 [1:0]	CNF12[1:0]	MODE12 [1:0]

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
CNF11[1:0]	MODE11 [1:0]	CNF10[1:0]	MODE10 [1:0]	CNF9[1:0]	MODE9[1:0]	CNF8[1:0]	MODE8[1:0]

Bit	Name	Access	Description	Reset value
[31:30] [27:26] [23:22] [19:18] [15:14] [11:10] [7:6] [3:2]	CNFy[1:0]	RW	GPIOA(y=8-15), GPIOB(y=8-9), the configuration bits for port x, by which the corresponding port is configured. When in input mode (MODE=00b). 00: Analog input mode. 01: Floating input mode. 10: With pull-up and pull-down mode. 11: Reserved. In output mode (MODE>00b). 00: General-purpose push-pull output mode. 10: Alternate function push-pull output mode. (I2C automatic open drain)	01b
[29:28] [25:24] [21:20] [17:16] [13:12] [9:8] [5:4] [1:0]	MODEy[1:0]	RW	GPIOA(y=8-15), GPIOB(y=8-9), port x mode selection, configure the corresponding port by these bits. 00: Input mode. 01/10/11: Output mode, the maximum speed is 30MHz.	00b

6.3.1.3 Port Input Register (GPIOx_INDR) (x=A/B)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:0]	IDRy	RO	(y=0-15), port input data. These bits are read-only and can only be read in 16-bit form. The value read is the high and low status of the corresponding bit.	X

6.3.1.4 Port Output Register (GPIOx_OUTDR) (x=A/B)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:0]	ODRy	RW	For output mode: (y=0-15), the data output by the port. These data can only be manipulated in 16-bit form. The IO port outputs the values of these registers to the	0

			outside world. For input mode with pull-down input: 0: Pull-down input; 1: Pull-up input.	

6.3.1.5 Port Reset/Set Register (GPIOx_BSHR) (x=A/B)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0

Bit	Name	Access	Description	Reset value
[31:16]	BRy	WO	(y=0-15), setting these bits will clear the corresponding OUTDR bits, writing 0 will have no effect. These bits can only be accessed in 16-bit form. If both the BR and BS bits are set, the BS bit takes effect.	0
[15:0]	BSy	WO	y=0-15), setting these bits will cause the corresponding OUTDR bit to be set, writing 0 will have no effect. These bits can only be accessed in 16-bit form. If both the BR and BS bits are set, the BS bit takes effect.	0

6.3.1.6 Port Reset Register (GPIOx_BCR) (x=A/B)

Offset address: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:0]	BRy	WO	(y=0-15), setting these bits will clear the corresponding OUTDR bits, writing 0 will have no effect. These bits can only be accessed in 16-bit form.	0

6.3.1.7 Port Lock Configuration Register (GPIOx_LCKR) (x=A/B)

Offset address: 0x18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LCKK
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0

Bit	Name	Access	Description	Reset value
[31:17]	Reserved	RO	Reserved	0
16	LCKK	RW	Lock key, which can be locked by a specific sequence of writes, but it can be read out at any time. When it reads 0, it means that the lock is not in effect, and when it reads 1, it means that the lock is in effect. The writing sequence of the lock key is: write 1-write 0-write 1-read 0-read 1. The last step is not necessary, but it can be used to confirm that the lock key has been activated. Any errors while writing the sequence will not activate the lock, and the value of LCK[7:0] cannot be changed while the sequence is being written. Once the lock is in effect, the port's configuration cannot be changed until the next reset.	0
[15:0]	LCKy	RW	(y=0-15), when these bits are 1, it means locking the configuration of the corresponding port. These bits can only be changed before LCKK is unlocked. The locked configuration refers to the configuration registers GPIOx_CFGLR and GPIOx_CFGHR.	0

Note: After the LOCK sequence is executed for the corresponding port bit, the configuration of the port bit will not be changed again until the next system reset.

6.3.2 AFIO Register Description

Unless otherwise specified, AFIO registers must be operated as words (operate these registers with 32 bits).

Table 6-12 List of AFIO-related registers

Name	Access address	Description	Reset value
R32_AFIO_PCFR1	0x40010004	Remap Register 1	0x00000000
R32_AFIO_EXTICR	0x40010008	External interrupt configuration register 1	0x00000000

6.3.2.1 Remap Register 1 (AFIO_PCFR1)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved					SWCFG[2:0]			Reserved					ADC_ETRGREG_RM	Reserved	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						TIM2_RM[1:0]		Reserved	TIM1_RM	Reserved	USART1_RM			I2C1_RM	

Bit	Name	Access	Description	Reset value
[31:27]	Reserved	RO	Reserved	0
[26:24]	SWCFG[2:0]	RW	These bits are used to configure the IO port of the SW function and tracking function. SWD (SDI) is the debug interface for accessing the core. After	0

			system reset, it always acts as SWD port. 0xx: Enable SWD (SDI); 100: Disable SWD (SDI), as GPIO function; Other: Invalid.	
[23:19]	Reserved	RO	Reserved	0
18	ADC_ETRGREG_RM	RW	Remap bit for ADC external trigger. 0: The ADC external trigger is connected to PA4; 1: The ADC external trigger is connected to PA15.	0
[17:10]	Reserved	RO	Reserved	0
[9:8]	TIM2_RM[1:0]	RW	Remap bit for Timer 2. These bits can be read and written by the user. It controls the mapping of channels 1 to 2 and 1N to 2N of Timer 2 on the GPIO port. 00: Default mapping (CH1/PA5, CH2/PA9, CH1N/PA7, CH2N/PA8); 01: Partial mapping (CH1/PA5, CH2/PA4, CH1N/PA7, CH2N/PA8); 10: Partial mapping (CH1/PA5, CH2/PA9, CH1N/PB5, CH2N/PB4); 11: Fully mapped (CH1/PA5, CH2/PA4, CH1N/PB5, CH2N/PB4).	0
7	Reserved	RO	Reserved	0
6	TIM1_RM	RW	Remap bit for Timer 1. These bits can be read and written by the user. It controls the mapping of channels 1 to 3, 1N to 3N, external trigger (ETR) and brake input (BKIN) of timer 1 on the GPIO port. 0: Default mapping (ETR/PB9, CH1/PB2, CH2/PB3, CH3/PB4, BKIN/PA15, CH1N/PA2, CH2N/PA3, CH3N/PA4); 1: Remapping (ETR/PB9, CH1/PB2, CH2/PB3, CH3/PA2, BKIN/PA15, CH1N/PB4, CH2N/PB5, CH3N/PA4);	0
5	Reserved	RO	Reserved	0
[4:2]	USART1_RM	RW	USART1 map configuration: 000: Default mapping (TX/PA11, RX/PA12, CTS/PA13, RTS/PA14); 001: Remapping (TX/PB1, RX/PB0, CTS/PA13, RTS/PA14); 010: Remapping (TX/PAO, RX/PA1, CTS/PA13, RTS/PA14); 011: Remapping (TX/PA1, RX/PA0, CTS/PA13, RTS/PA14). 100: Remapping (TX/PA2, RX/PA3, CTS/PA13, RTS/PA14).	0
[1:0]	I2C1_RM	RW	I2C1 Remapping: 00: Default mapping (SCL/PA13, SDA/PA14); 01: Remapping (SCL/PA2, SDA/PA3); 1X: Remapping (SCL/PA11, SDA/PA12).	0

6.3.2.2 External Interrupt Configuration Register 1 (AFIO_EXTICR)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI 15	EXTI 14	EXTI 13	EXTI 12	EXTI 11	EXTI 10	EXTI 9	EXTI 8	EXTI 7	EXTI 6	EXTI 5	EXTI 4	EXTI 3	EXTI 2	EXTI 1	EXTI 0

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	EXTIx	RW	External interrupt input pin configuration bit. Used to determine to which port pins the external interrupt pins are mapped. 0: xth pin of the PA pin, x=0-15; 1: xth pin of the PB pin, x=0-9; Others: Reserved.	0

Chapter 7 Direct Memory Access Control (DMA)

The Direct Memory Access (DMA) controller provides a high-speed means of data transfer between peripherals and memory or between memory and memory without CPU intervention, data can be moved quickly through the DMA to save CPU resources for other operations.

Each channel of the DMA controller is dedicated to managing requests for memory access from one or more peripherals. There is also an arbiter to co-ordinate priorities between the channels.

7.1 Main Features

- Multiple independently configurable channels
- Each channel is directly connected to a dedicated hardware DMA request and supports software triggering
- Support Buffer management with loop
- Request priority between multiple channels can be set by software programming (very high, high, medium and low) and priority setting is determined by the channel number when equal (the lower the channel number the higher the priority)
- Support peripheral-to-memory, memory-to-peripheral, and memory-to-memory transfers
- Flash memory, SRAM, peripheral SRAM and HB peripherals can be used as access sources and targets
- Programmable number of data transfer bytes: up to 65535

7.2 Function Description

7.2.1 DMA Channel Processing

1) Arbitration priority

DMA requests generated by multiple independent channels are fed to the DMA controller via a logical or structure, and only one channel request is currently responded to. An arbiter inside the module selects the peripheral/memory access to be initiated based on the priority of the channel request.

In software management, the application can configure the priority level for each channel independently by setting the PL[1:0] bits of the DMA_CFGRx register, including 4 levels: highest, high, medium and low. When the software setting levels are the same between channels, the module will be selected according to a fixed hardware priority, with the lower channel number having a higher priority than the higher one.

2) DMA configuration

When the DMA controller receives a request signal, it accesses the requested peripheral or memory and establishes a data transfer between the peripheral or memory and the memory. It consists of the following 3 main operation steps.

(1) Fetch data from the memory address indicated by the Peripheral Data Register or the Current Peripheral/Memory Address Register. The start address for the first transfer is the peripheral base address or memory address specified by the DMA_PADDRx or DMA_MADDRx registers.

(2) Store data to the memory address indicated by the Peripheral Data Register or the Current Peripheral/Memory Address Register, and the start address for the first transfer is the peripheral base address or memory address specified by the DMA_PADDRx or DMA_MADDRx registers.

(3) Performs a decrement operation of the value in the DMA_CNTRx register, which indicates the number of operations currently outstanding for transfer.

Each channel includes 3 types of DMA data transfer methods.

- Peripheral to memory (MEM2MEM=0, DIR=0)
- Memory to peripheral (MEM2MEM=0, DIR=1)
- Memory to memory (MEM2MEM=1)

Note: The memory-to-memory mode does not require a peripheral request signal. After configuring this mode (MEM2MEM=1), the channel is turned on (EN=1) to start data transfer. This mode does not support cyclic mode.

The configuration process is as follows.

1) Set the first address of the peripheral register or the memory data address in the memory-to-memory mode (MEM2MEM=1) in the DMA_PADDRx register. This address will be the source or destination address for data transfer when a DMA request occurs.
2) Set the memory data address in the DMA_MADDRx register. When a DMA request occurs, the transferred data will be read from or written to this address.
3) Set the amount of data to be transferred in the DMA_CNTRx register. This value is decremented after each data transfer.
4) Set the priority of the channel in the PL[1:0] bits of the DMA_CFGRx register.
5) Set the direction of data transfer, cyclic mode, incremental mode for peripheral and memory, data width for peripheral and memory, transfer halfway, transfer complete, and transfer error interrupt enable bits in the DMA_CFGRx register.
6) Set the ENABLE bit of the DMA_CCRx register to start channel x.

Note: The DMA_PADDRx/DMA_MADDRx/DMA_CNTRx registers and the direction of data transfer (DIR), cyclic mode (location), and incremental mode of peripherals and memory (MINC/PINC) control bits in the DMA_CFGRx register can be configured to write only when the DMA channel is turned off.

3) Circular mode

Setting CIRC position 1 of the DMA_CFGRx register enables the cyclic mode function for channel data transfers. In cyclic mode, when the number of data transfers becomes 0, the contents of the DMA_CNTRx register are automatically reloaded to its initial value, and the internal peripheral and memory address registers are reloaded to the initial address values set by the DMA_PADDRx and DMA_MADDRx registers, and DMA operation will continue until the channel is turned off or the DMA mode is turned off.

4) DMA processing status

- Transfer half: It corresponds to the hardware setting of HTIFx bit in DMA_INTFR register. The DMA transfer bytes half flag will be generated when the number of DMA transfers is reduced to less than half of the initial set value, and an interrupt will be generated if HTIE is set in the DMA_CCRx register. The hardware uses this flag to alert the application that it can prepare for a new round of data transfers.
- Transfer completion: corresponds to the hardware setting of the TCIFx bit in the DMA_INTFR register. When the number of DMA transfer bytes decreases to 0, the DMA transfer completion flag will be generated, and if TCIE is set in the DMA_CCRx register, an interrupt will be generated.
- Transfer error: corresponds to a hardware set of the TEIFx bit in the DMA_INTFR register. Reading and writing a reserved address area will generate a DMA transfer error. At the same time the module hardware will automatically clear the EN bit of the DMA_CCRx register corresponding to the channel where the error occurred, and the channel is turned off. If TEIE is set in the DMA_CCRx register, an interrupt will be generated.

When the application queries the DMA channel status, it can first access the GIFx bit of the DMA_INTFR register

to determine which channel is currently experiencing a DMA event, and then process the specific DAM event content for that channel.

7.2.2 Programmable Total Data Transfer Size/Data Bit Width/Alignment

The total size of the data to be transferred per DMA channel round is programmable up to 65535 times, and the number of pending transfer bytes is indicated in the DMA_CNTRx register. At EN=0, the set value is written, and at EN=1 when the DMA transfer channel is turned on, this register becomes a read-only attribute with a decreasing value after each transfer.

The transferred data fetch values of peripherals and memories support the address pointer auto-increment function with programmable pointer increments. The first transmitted data address they access is stored in the DMA_PADDRx and DMA_MADDRx registers.By setting the PINC bit or MINC position 1 of the DMA_CFGRx register, the peripheral address self-increment mode or memory address self-increment mode can be enabled, respectively. PSIZE[1:0] sets the peripheral address fetch data size and address self-increment size. MSIZE[1:0] sets the memory address to take the data size and address self-increasing small, including three choices: 8-bit, 16-bit, 32-bit. The specific data transfer methods are listed in the following table.

Table 7-1 DMA transfer with different data bit widths (PINC=MINC=1)

Source bit width	Objectives bit width	Transmission number	Source: address/data	Target: address/data	Transfer operations
8	8	4	0x00/B0 0x01/B1 0x02/B2 0x03/B3	0x00/B0 0x01/B1 0x02/B2 0x03/B3	● The source address increment is aligned with the data bit width set at the source and takes a value equal to the data bit width at the source ● The target address increment is aligned with the bit width of the target setup data and takes a value equal to the target data bit width ● DMA transfer of data sent to the target based on the principle: the high bit of the data size is not enough to make up 0, the high bit of the data size overflow is removed ● Storage data mode: small-end mode, low address stores low bytes, high address stores high bytes
8	16	4	0x00/B0 0x01/B1 0x02/B2 0x03/B3	0x00/00B0 0x02/00B1 0x04/00B2 0x06/00B3	
8	32	4	0x00/B0 0x01/B1 0x02/B2 0x03/B3	0x00/000000B0 0x04/000000B1 0x08/000000B2 0x0C/000000B3	
16	8	4	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	0x00/B0 0x01/B2 0x02/B4 0x03/B6	
16	16	4	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	
16	32	4	0x00/B1B0 0x02/B3B2 0x04/B5B4 0x06/B7B6	0x00/0000B1B0 0x04/0000B3B2 0x08/0000B5B4 0x0C/0000B7B6	
32	8	4	0x00/B3B2B1B0 0x04/B7B6B5B4 0x08/BBBAB9B8 0x0C/BFBEBDBC	0x00/B0 0x01/B4 0x02/B8 0x03/BC	
32	16	4	0x00/B3B2B1B0 0x04/B7B6B5B4 0x08/BBBAB9B8 0x0C/BFBEBDBC	0x00/B1B0 0x02/B5B4 0x04/B9B8 0x06/BDBC	
32	32	4	0x00/B3B2B1B0 0x04/B7B6B5B4	0x00/B3B2B1B0 0x04/B7B6B5B4	

			0x08/BBBAB9B8 0x0C/BFBEBDBC	0x08/BBBAB9B8 0x0C/BFBEBDBC	

7.2.3 DMA Request Mapping

Figure 7-1 DMA request image

Table 7-2 DMA1 peripheral mapping table for each channel

Peripheral	Channel1	Channel 2	Channel 3	Channel 4	Channel 5	Channel 6	Channel 7
ADC	ADC						
USART1				USART1_TX	USART1_RX		
I2C1						I2C1_TX	I2C1_RX
TIM1		TIM1_CH1	TIM1_CH2	TIM1_TRIG TIM1_COM	TIM1_UP	TIM1_CH3	

7.3 Register Description

Table 7-3 DMA-related registers list

Name	Access address	Description	Reset value
R32_DMA_INTFR	0x40020000	DMA interrupt status register	0x00000000
R32_DMA_INTFCR	0x40020004	DMA interrupt flag clear register	0x00000000
R32_DMA_CFGR1	0x40020008	DMA channel 1 configuration register	0x00000000
R32_DMA_CNTR1	0x4002000C	DMA channel 1 transfer data number register	0x00000000
R32_DMA_PADDR1	0x40020010	DMA channel 1 peripheral address register	0x00000000
R32_DMA_MADDR1	0x40020014	DMA channel 1 memory address register	0x00000000
R32_DMA_CFGR2	0x4002001C	DMA channel 2 configuration register	0x00000000
R32_DMA_CNTR2	0x40020020	DMA channel 2 transfer data number register	0x00000000
R32_DMA_PADDR2	0x40020024	DMA channel 2 peripheral address register	0x00000000
R32_DMA_MADDR2	0x40020028	DMA channel 2 memory address register	0x00000000
R32_DMA_CFGR3	0x40020030	DMA channel 3 configuration register	0x00000000
R32_DMA_CNTR3	0x40020034	DMA channel 3 transfer data number register	0x00000000
R32_DMA_PADDR3	0x40020038	DMA channel 3 peripheral address register	0x00000000
R32_DMA_MADDR3	0x4002003C	DMA channel 3 memory address register	0x00000000
R32_DMA_CFGR4	0x40020044	DMA channel 4 configuration register	0x00000000
R32_DMA_CNTR4	0x40020048	DMA channel 4 transfer data number register	0x00000000
R32_DMA_PADDR4	0x4002004C	DMA channel 4 peripheral address register	0x00000000
R32_DMA_MADDR4	0x40020050	DMA channel 4 memory address register	0x00000000
R32_DMA_CFGR5	0x40020058	DMA channel 5 configuration register	0x00000000
R32_DMA_CNTR5	0x4002005C	DMA channel 5 transfer data number register	0x00000000
R32_DMA_PADDR5	0x40020060	DMA channel 5 peripheral address register	0x00000000
R32_DMA_MADDR5	0x40020064	DMA channel 5 memory address register	0x00000000
R32_DMA_CFGR6	0x4002006C	DMA channel 6 configuration register	0x00000000
R32_DMA_CNTR6	0x40020070	DMA channel 6 transfer data number register	0x00000000
R32_DMA_PADDR6	0x40020074	DMA channel 6 peripheral address register	0x00000000
R32_DMA_MADDR6	0x40020078	DMA channel 6 memory address register	0x00000000
R32_DMA_CFGR7	0x40020080	DMA channel 7 configuration register	0x00000000
R32_DMA_CNTR7	0x40020084	DMA channel 7 transfer data number register	0x00000000
R32_DMA_PADDR7	0x40020088	DMA channel 7 peripheral address register	0x00000000
R32_DMA_MADDR7	0x4002008C	DMA channel 7 memory address register	0x00000000

7.3.1 DMA Interrupt Status Register (DMA_INTFR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1

Bit	Name	Access	Description	Reset value
[31:28]	Reserved	RO	Reserved	0
27/23/19/15/11/7/3	TEIFx	RO	Transmission error flag for channel x (x=1/2/3/4/5/6/7). 1: A transmission error occurred on channel x. 0: No transmission error on channel x. Hardware set, software write CTEIFx bit to clear this flag.	0
26/22/18/1	HTIFx	RO	Transmission halfway flag for channel x (x=1/2/3/4/5/6/7).	0

4/10/6/2			1: A transmission over half event is generated on channel x. 0: No transmission over half on channel x. Hardware set, software write CHTIFx bit to clear this flag.	
25/21/17/13/9/5/1	TCIFx	RO	Transmission completion flag for channel x (x=1/2/3/4/5/6/7). 1: A transmission completion event is generated on channel x. 0: No transmission completion event on channel x. Hardware set, software write CTCIFx bit to clear this flag.	0
24/20/16/12/8/4/0	GIFx	RO	Global interrupt flag for channel x (x=1/2/3/4/5/6/7). 1: TEIFx or HTIFx or TCIFx is generated on channel x. 0: No TEIFx or HTIFx or TCIFx occurred on channel x. Hardware set, software write CGIFx bit to clear this flag.	0

7.3.2 DMA Interrupt Flag Clear Register (DMA_INTFCR)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved				CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1

Bit	Name	Access	Description	Reset value
[31:28]	Reserved	RO	Reserved	0
27/23/19/15/11/7/3	CTEIFx	WO	Clear the transmission error flag for channel x (x=1/2/3/4/5/6/7). 1: Clear the TEIFx flag in the DMA_INTFR register. 0: No effect.	0
26/22/18/14/10/6/2	CHTIFx	WO	Clear the transmission halfway flag for channel x (x=1/2/3/4/5/6/7). 1: Clear the HTIFx flag in the DMA_INTFR register. 0: No effect.	0
25/21/17/13/9/5/1	CTCIFx	WO	Clear the transmission completion flag for channel x (x=1/2/3/4/5/6/7). 1: Clear the TCIFx flag in the DMA_INTFR register. 0: No effect.	0
24/20/16/12/8/4/0	CGIFx	WO	Clear the global interrupt flag for channel x (x=1/2/3/4/5/6/7). 1: Clear the TEIFx/HTIFx/TCIFx/ GIFx flags in the DMA_INTFR register. 0: No effect.	0

7.3.3 DMA Channel x Configuration Register (DMA_CFGRx)(x=1/2/3/4/5/6/7)

Offset address: 0x08 + (x-1)*20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN

Bit	Name	Access	Description	Reset value
[31:15]	Reserved	RO	Reserved	0

14	MEM2MEM	RW	Memory-to-memory mode enable. 1: Enable memory-to-memory data transfer mode. 0: Disable memory-to-memory data transfer mode.	0
[13:12]	PL[1:0]	RW	Channel priority setting. 00: low; 01: medium. 10: High; 11: Very high.	0
[11:10]	MSIZE[1:0]	RW	Memory address data width setting. 00: 8-bit; 01: 16-bit. 10: 32-bit; 11: Reserved.	0
[9:8]	PSIZE[1:0]	RW	Peripheral address data width setting. 00: 8-bit; 01: 16-bit. 10: 32-bit; 11: Reserved.	0
7	MINC	RW	Memory address incremental increment mode enable. 1: Enable incremental memory address increment operation. 0: Memory address remains unchanged operation.	0
6	PINC	RW	Peripheral address incremental increment mode enable. 1: Enable incremental increment operation of the peripheral address. 0: Peripheral address remains unchanged operation.	0
5	CIRC	RW	DMA channel cyclic mode enable. 1: Enables cyclic operation. 0: Perform a single operation.	0
4	DIR	RW	Data transfer direction. 1: Read from memory. 0: Read from peripheral.	0
3	TEIE	RW	Transmission error interrupt enable control. 1: Enable transmission error interrupt. 0: Disable transmission error interrupt.	0
2	HTIE	RW	Transmission over half interrupt enable control. 1: Enable the transmission over half interrupt. 0: Disable the transmission over half interrupt.	0
1	TCIE	RW	Transmission completion interrupt enable control. 1: Enable the transmission completion interrupt. 0: Disable the transmission completion interrupt.	0
0	EN	RW	Channel enable control. 1: Channel on; 0: Channel off. When a DMA transfer error occurs, the hardware automatically clears this bit to 0 and shuts down the channel.	0

7.3.4 DMA Channel x Number of Data Register (DMA_CNTRx) (x=1/2/3/4/5/6/7)

Offset address: 0x0C + (x-1)*20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NDT[15:0]															

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:0]	NDT[15:0]	RW	Number of data transfers, range 0-65535. This register can only be written when the channel is not operating (EN=0 for DMA_CFGRx). After the channel is	0

			turned on this register becomes read-only and indicates the number of remaining pending transfer bytes (the register content is decremented after each DMA transfer). When the channel is in cyclic mode, the contents of the register will be automatically reloaded to the previously configured value.	

Note: This register can only be changed when EN=0; when EN=1, it is a read-only register, indicating the current number of pending transfer bytes. When the register content is 0, no data transmission will occur regardless of whether the channel is on or off.

7.3.5 DMA Channel x Peripheral Address Register (DMA_PADDRx) (x=1/2/3/4/5/6/7)

Offset address: 0x10 + (x-1)*20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PA[31:0]																															

Bit	Name	Access	Description	Reset value
[31:0]	PA[31:0]	RW	Peripheral base address, which serves as the source or destination address for peripheral data transfer. When PSIZE[1:0]='01' (16 bits), the module automatically ignores bit0 and the operation address is automatically 2-byte aligned; when PSIZE[1:0]='10' (32 bits), the module automatically ignores bit[1:0] and the operation address is automatically 4-byte aligned.	0

Note: This register can only be changed when EN=0 and cannot be written when EN=1.

7.3.6 DMA Channel x Memory Address Register (DMA_MADDRx) (x=1/2/3/4/5/6/7)

Offset address: 0x14 + (x-1)*20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MA[31:0]																															

Bit	Name	Access	Description	Reset value
[31:0]	MA[31:0]	RW	The memory data address, which serves as the source or destination address for data transfers. When MSIZE[1:0]='01' (16 bits), the module automatically ignores bit0, and the operation address is automatically 2-byte aligned; when MSIZE[1:0]='10' (32 bits), the module automatically ignores bit[1:0], and the operation address is automatically 4-byte aligned.	0

Note: This register can only be changed when EN=0 and cannot be written when EN=1.

Chapter 8 Analog-to-digital Converter (ADC)

The ADC module contains one 10-bit successive approximation ADC, allowing an input clock of up to 12MHz. Supports 15 external channels and 1 internal signal source sampling source. It can complete single conversion, continuous conversion of channels, automatic scanning mode between channels, intermittent mode, external trigger mode, trigger delay and other functions. The analog watchdog function can be used to monitor whether the channel voltage is within the threshold range.

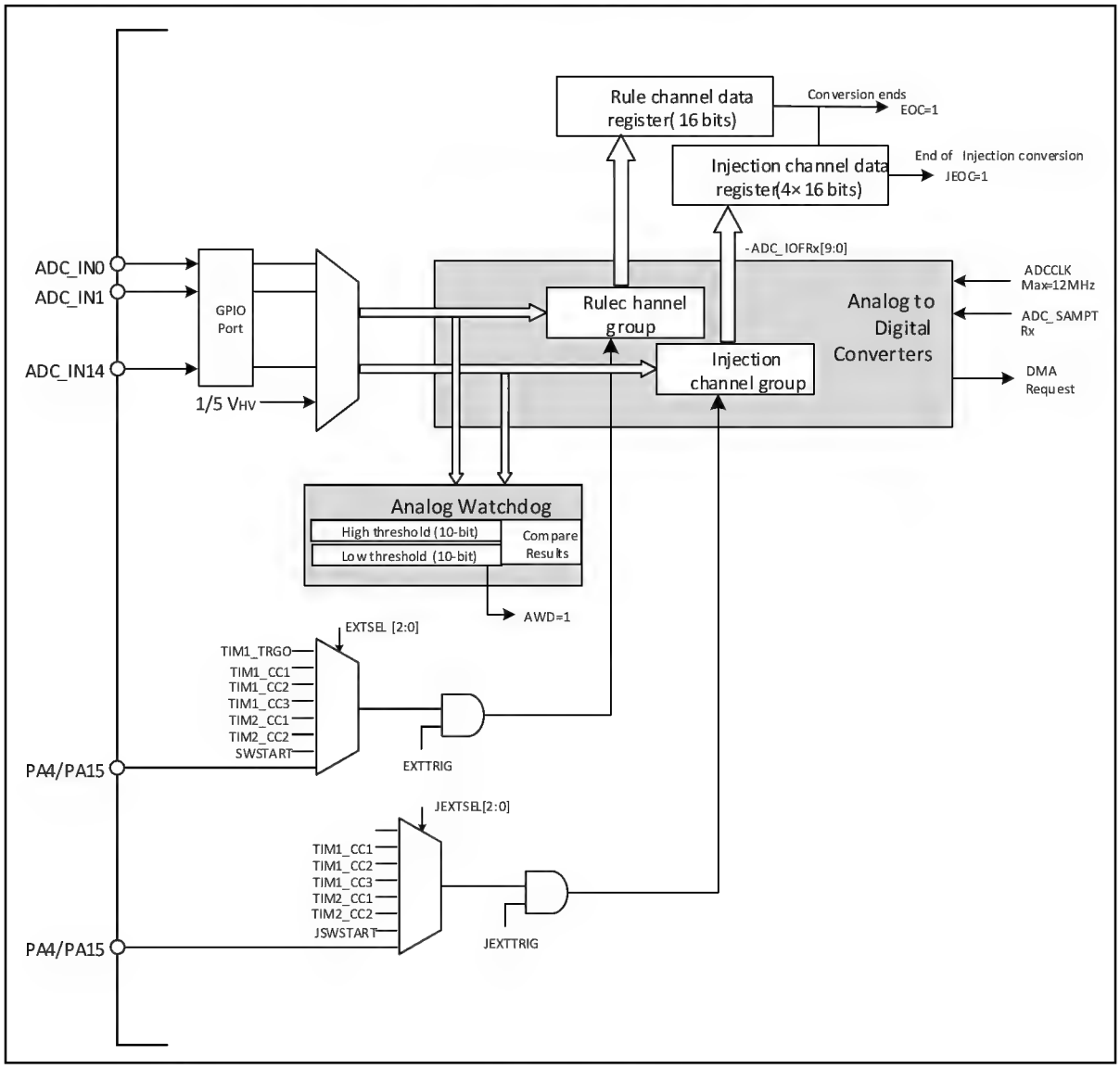
8.1 Main Features

- 10-bit resolution
- Supports 15 external channels and 1 internal signal sources for sampling
- Multiple sampling conversion methods for multiple channels: single, continuous, scan, trigger, intermittent, etc.
- Data alignment modes: left-aligned, right-aligned
- Sampling time can be programmed separately by channel
- Both rule conversion and injection conversion support external triggering
- Analog watchdog to monitor channel voltage, self-calibration function
- ADC channel input range: $GND \leq V_{IN} \leq V_{33}$
- Trigger delay

8.2 Functional Description

8.2.1 Module Structure

Figure 8-1 ADC module block diagram



8.2.2 ADC Configuration

1) Module power-up

An ADON bit of 1 in the ADC_CTLR2 register indicates that the ADC module is powered up. When the ADC module enters the power-up state (ADON=1) from the power-down mode (ADON=0), a delay period t_{STAB} is required for the module stabilization time. After that, the ADON bit is written to 1 again and is used as the start signal for software to start the ADC conversion. By clearing the ADON bit to 0, the current conversion can be terminated and the ADC module placed in power-down mode, a state in which the ADC consumes almost no power.

2) Sampling clock

The module's register operation is based on the HBCLK (HB bus) clock, and the clock reference ADCCLK of its conversion unit is divided by the ADCPRE field configuration of the RCC_CFGR0 register. For detailed

information, refer to the data sheet CH641DS0.

3) Channel configuration

The ADC module provides 14 channel sampling sources. They can be configured into two types of conversion groups: regular groups and injection groups. to achieve a group conversion consisting of a series of conversions in any order on any number of channels.

Conversion group:

- Rule group: consists of up to 16 conversions. The rule channels and their conversion order are set in the ADC_RSQRx register. The total number of conversions in the rule group should be written to L[3:0] in the ADC_RSQR1 register.
- Injection group: consists of up to 4 conversions. The injection channels and the order of their conversions are set in the ADC_ISQR register. The total number of conversions in the injection group should be written in JL[1:0] of the ADC_ISQR register.

Note: (1) If the ADC_RSQRx or ADC_ISQR registers are changed during conversion, the current conversion is terminated and a new start signal is sent to the ADC to convert the newly selected group.

(2) ADC_IN9 connects to the output of QII_OP.

1 internal channel:

- 1/5 divider of V_{HV} voltage: Connect the ADC_IN15 channel and the internal channel sampling time is recommended to be set above 40us.

4) Calibration

The ADC has a built-in self-calibration mode. The accuracy error caused by changes in the internal capacitor bank can be greatly reduced through the calibration process. During calibration, an error correction code is calculated on each capacitor, which is used to eliminate errors produced on each capacitor in subsequent conversions.

Initialize the calibration register by writing the RSTCAL bit of the ADC_CTLR2 register to 1, and wait for the RSTCAL hardware to clear to 0 to indicate that the initialization is complete. Set the CAL bit to start the calibration function. Once the calibration is completed, the hardware will automatically clear the CAL bit and store the calibration code in ADC_RDATAR. After that normal conversion functions can begin. It is recommended to perform an ADC calibration when the ADC module is powered on.

Note: Before starting calibration, you must ensure that the ADC module is in the power-on state (ADON=1) for at least two ADC clock cycles.

5) Programmable sampling time

The ADC uses several ADCCLK cycles to sample the input voltage. The number of sampling cycles for a channel can be changed using the SMPx[1:0] bits in the ADC_SAMPTR1 and ADC_SAMPTR2 registers. Each channel can be sampled separately using a different time.

The total conversion time is calculated as follows.

$$T_{CONV} = \text{sampling time} + 11T_{ADCCLK}$$

The ADC's rule channel conversion supports the DMA function. The value of the rule channel conversion is stored in a data-only register, ADC_RDATAR. To prevent the data in ADC_RDATAR register from being fetched in time when multiple rule channels are converted in succession, the DMA function of ADC can be enabled. The hardware will generate a DMA request at the end of the conversion of a rule channel (EOC set) and transfer the converted data from the ADC_RDATAR register to the user-specified destination address.

After the channel configuration of the DMA controller module is completed, write DMA position 1 of the

ADC_CTLR2 register to enable the DMA function of the ADC.

Note: Injection group conversion does not support DMA function.

6) Data alignment

The ALIGN bit in the ADC_CTLR2 register selects the alignment of the ADC converted data storage. 12-bit data supports left-aligned and right-aligned modes.

The data register ADC_RDATAR of the rule group channel holds the actual converted 12-bit digital value; while the data register ADC_IDATARx of the injection group channel is the actual converted data minus the value written after the offset defined in the ADC_IOFRx register, there will be positive and negative cases, so there are sign bits (SIGNB).

Figure 8-1 Data left alignment

Rule group data register

D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0	0	0	0

Inject group data register

SIGNB	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0	0	0

Figure 8-2 Data right alignment

Rule group data register

0	0	0	0	0	0	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Inject group data register

SIGNB	SIGNB	SIGNB	SIGNB	SIGNB	SIGNB	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

8.2.3 External Trigger Source

The start event of ADC conversion can be triggered by external events. If the EXTTRIG or JEXTTRIG bits of the ADC_CTLR2 register are set, conversions of rule group or injection group channels can be triggered by external events, respectively. At this time, the configuration of the EXTSEL[2:0] and JEXTSEL[2:0] bits determines the external event source of the rule group and injection group. ADC_TRIG_SEL0-4 bits configure the comparison event of TIM1 and TIM2 as an external trigger event for the injection group or the rule group.

Note: When the external trigger signal is selected for ADC rule or injection conversion, only its rising edge can initiate conversion.

Table 8-3 External trigger sources for rule group channels

EXTSEL[2:0]	Trigger source	Type
000	TRGO event of timer 1	Internal signal from on-chip timer
001	CC1 event of timer 1	
010	CC2 event of timer 1	
011	CC3 event of timer 1	
100	CC1 event of timer 2	
101	CC2 event of timer 2	
110	PA4/PA15 event	From external pins
111	SWSTART software trigger	Software control bits

Table 8-4 External trigger sources for injection group channels

JEXTSEL[2:0]	Trigger source	Type

000	-	Internal signal from on-chip timer
001	CC1 event of timer 1	
010	CC2 event of timer 1	
011	CC3 event of timer 1	
100	CC1 event of timer 2	
101	CC2 event of timer 2	
110	PA4/PA15 event	From external pins
111	JSWSTART software trigger	Software control bits

8.2.4 Conversion Mode

Table 8-5 Conversion mode combinations

ADC_CTLR1 and ADC_CTLR2 register control bits					ADC conversion mode
CONT	SCAN	DISCEN/JDISCEN	JAUTO	Start event	
0	0	0	0	ADON position 1	Single single-channel mode: A rule channel performs a single conversion.
				External trigger method	Single single-channel mode: A single conversion is performed on one of the rule channels or injection channels.
	1	0	0	ADON position 1 or external trigger method	Single scan mode: performs a single conversion of all selected rule group channels (ADC_RSQRx) or all injection group channels (ADC_ISQR) one by one in sequence. Trigger injection method: When the rule group channel conversion process can be inserted into the injection group channel all conversion, and then continue the rule group channel conversion afterwards; but the rule group channel conversion will not be inserted when converting the injection group channel.
			1	ADON position 1 or external trigger method	Single scan mode: performs a single conversion of all selected rule group channels (ADC_RSQRx) or all injection group channels (ADC_ISQR) one by one in sequence. Automatic injection method: After the rule group channel is converted, the injection group channel is automatically converted. *Note: External trigger signals injected into the channel are not allowed during the conversion process.*
	0	1 (DISCEN and JDISCEN cannot be 1 at the same time)	0	External trigger method	Single intermittent mode: Each time an event is started, a short sequence (DISCNUM[2:0] defined number) of channel number transitions is executed and cannot be restarted until all selected channel transitions are completed. *Note: The DISCEN and JDISCEN control bits are selected for the rule group and injection group respectively, and the intermittent mode cannot be configured for the rule group and injection group at the same time.*
			1	-	Disable this mode.
	1	1	X	-	No such mode.
1	0	0	0	ADON position 1 or external trigger	Continuous single channel/scan mode: repeat a new round of transitions at the end of each round until CONT clears 0 to terminate.
	1	0	0		
			1		

				method	

Note: The external trigger events for rule groups and injection groups are different, and the 'ACON' bit can only initiate rule group channel conversion, so the initiation events for rule group and injection group channel conversion are independent.

1) Single single-channel conversion mode

In this mode, only one conversion is executed for the current 1 channel. This mode performs conversion for the channel that is sorted 1st in the rule group or injection group, where it is initiated by setting ADON position 1 of the ADC_CTLR2 register (for rule channels only) or can be initiated by external trigger (for rule channels or injection channels). Once the conversion of the selected channel is completed it will.

If the conversion is for a rule group channel, the conversion data is stored in the 16-bit ADC_RDATAR register, the EOC flag is set, and an ADC interrupt is triggered if the EOCIE bit is set.

If the conversion is for an injection group channel, the conversion data is stored in the 16-bit ADC_IDATAR1 register, the EOC and JEOC flags are set, and an ADC interrupt is triggered if the JEOCIE or EOCIE bit is set.

2) Single scan mode conversion

The ADC scan mode is entered by setting the SCAN bit of the ADC_CTLR1 register to 1. This mode is used to scan a group of analog channels and perform a single conversion for all channels selected by ADC_RSQRx register (for regular channels) or ADC_ISQR (for injection channels) one by one, and the next channel in the same group is converted automatically when the current channel conversion is finished.

In the scan mode, there is a subdivision into triggered injection mode and automatic injection mode depending on the status of the JAUTO bit.

- Trigger injection

JAUTO bit is 0. When the trigger event of injection group channel conversion occurs during the scanning of rule group channels, the current conversion is reset and the sequence of injection channels is performed in a single scan, and the last interrupted rule group channel conversion is resumed after all selected injection group channel scanning conversions are completed.

If a rule channel start event occurs while the injection group channel sequence is currently being scanned, the injection group conversion is not interrupted, but the rule sequence conversion is executed again after the injection sequence conversion is completed.

Note: When using triggered injection conversions, you must ensure that the interval between triggered events is longer than the injection sequence. For example, if the overall time to complete the conversion of the injection sequence takes 28 ADCCLK, then the minimum value of the event interval to trigger the injection channel is 29 ADCCLK.

- Auto-injection

The JAUTO bit is set to 1, and conversion of the selected channel of the injection group is performed automatically after scanning all the channels selected by the rule group for conversion. This approach can be used to convert up to 20 conversion sequences in the ADC_RSQRx and ADC_ISQR registers.

In this mode, external triggering of the injection channel must be disabled (JEXTTRIG=0).

Note: When the ADC clock prescaler coefficient (ADCPRE[1:0]) is 4 to 8, when switching from regular conversion to injection sequence or from injection conversion to regular sequence, 1 ADCCLK interval will be automatically inserted; When the ADC clock prescaler factor is 2, there is a delay of 2 ADCCLK intervals.

3) Single intermittent mode conversion

The intermittent mode of the rule group or injection group is entered by setting the DISCEN or JDISCEN bit of the

ADC_CTLR1 register to 1. This mode differs from scanning a complete set of channels in scan mode, but divides a set of channels into multiple short sequences, and each external trigger event will perform a short sequence of scan transitions.

The length n (n<=8) of the short sequence is defined in DISCNUM[2:0] of the ADC_CTLR1 register. When DISCEN is 1, it is the discontinuous mode of the rule group, and the total length to be converted is defined in L[3:0] of the ADC_RSQR1 register; when JDISCEN is 1, it is the discontinuous mode of the injection group, and the total length to be converted is defined in JL[1:0] of the ADC_ISQR register. You cannot set both a rule group and an injection group to discontinuous mode at the same time.

Example of rule group intermittent mode.

DISCEN=1, DISCNUM[2:0]=3, L [3:0]=8, channels to be converted = 1, 3, 2, 5, 8, 4, 10, 6

The 1st external trigger: conversion sequence is: 1, 3, 2

The 2nd external trigger: conversion sequence is: 5, 8, 4

The 3rd external trigger: conversion sequence is: 10, 6, while generating EOC events

The 4th external trigger: conversion sequence is: 1, 3, 2

Examples of intermittent patterns injected into groups.

JDISCEN=1, DISCNUM[2:0]=1, JL[1:0]=3, channel to be converted=1, 3, 2

The 1st external trigger: conversion sequence is: 1

The 2nd external trigger: the conversion sequence is: 3

The 3rd external trigger: conversion sequence is: 2, generating both EOC and JEOC events

The 4th external trigger: conversion sequence is: 1

Note: 1. When converting a rule group or injection group in intermittent mode, the conversion sequence does not automatically start from the beginning when it ends. When all subgroups have been converted, the next trigger event starts the conversion of the first subgroup.

2. You cannot use auto-injection (JAUTO=1) and intermittent mode at the same time.

3. You cannot set intermittent mode for both rule groups and injection groups, and intermittent mode can only be used for a group of conversions.

4) Continuous conversion

In the continuous conversion mode, another conversion is started as soon as the previous ADC conversion is completed. The conversion will not stop on the last channel of the selection group, but will continue conversion from the first channel of the selection group again. The startup events in this mode include external trigger events and the ADON bit is set to 1. After setting the startup, the CONT bit needs to be set to 1.

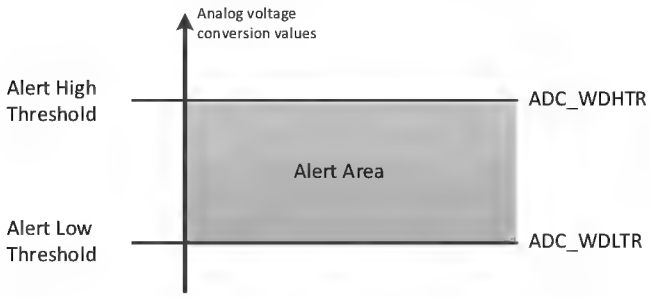
If a regular channel is converted, the conversion data is stored in the ADC_RDATAR register, the conversion end flag EOC is set, and if EOCIE is set, an interrupt is generated.

If an injection channel is converted, the conversion data is stored in the ADC_IDATARx register, the injection conversion end flag JEOC is set, and if JEOCIE is set, an interrupt is generated.

8.2.5 Analog Watchdog

The AWD analog watchdog status bit is set if the analog voltage being converted by the ADC is below the low threshold or above the high threshold. The threshold settings are located in the lowest 12 valid bits of the ADC_WDHTR and ADC_WDLTR registers. The AWDIE bit of the ADC_CTLR1 register is set to allow the corresponding interrupt to be generated.

Figure 8-2 Analog watchdog threshold area



Configure the AWDSGL, AWDEN, JAWDEN and AWDCH[4:0] bits of the ADC_CTLR1 register to select the channel for analog watchdog alerting, as related in the following table.

Table 8-6 Analog Watchdog channel selection

Analog Watchdog alert channel	ADC_CTLR1 register control bit			
	AWDSGL	AWDEN	JAWDEN	AWDCH[4:0]
No vigilance	Ignore	0	0	Ignore
All injection channels	0	0	1	Ignore
All rule channels	0	1	0	Ignore
All injection and rule channels	0	1	1	Ignore
Single injection channel	1	0	1	Determine the channel number
Single rule channel	1	1	0	Determine the channel number
Single injection and rule channel	1	1	1	Determine the channel number

8.3 Register Description

Table 8-7 ADC-related registers list

Name	Access address	Description	Reset value
R32_ADC_STATR	0x40012400	ADC status register	0x00000000
R32_ADC_CTLR1	0x40012404	ADC control register 1	0x00000000
R32_ADC_CTLR2	0x40012408	ADC control register 2	0x00000000
R32_ADC_SAMPTR1	0x4001240C	ADC sample time configuration register 1	0x00000000
R32_ADC_SAMPTR2	0x40012410	ADC sample time configuration register 2	0x00000000
R32_ADC_IOFR1	0x40012414	ADC injected channel data offset register 1	0x00000000
R32_ADC_IOFR2	0x40012418	ADC injected channel data offset register 2	0x00000000
R32_ADC_IOFR3	0x4001241C	ADC injected channel data offset register 3	0x00000000
R32_ADC_IOFR4	0x40012420	ADC injected channel data offset register 4	0x00000000
R32_ADC_WDHTR	0x40012424	ADC watchdog high threshold register	0x000003FF
R32_ADC_WDLTR	0x40012428	ADC watchdog low threshold register	0x00000000
R32_ADC_RSQR1	0x4001242C	ADC regular sequence register 1	0x00000000
R32_ADC_RSQR2	0x40012430	ADC regular sequence register 2	0x00000000
R32_ADC_RSQR3	0x40012434	ADC regular sequence register 3	0x00000000
R32_ADC_ISQR	0x40012438	ADC injected sequence register	0x00000000
R32_ADC_IDATAR1	0x4001243C	ADC injected data register 1	0x00000000
R32_ADC_IDATAR2	0x40012440	ADC injected data register 2	0x00000000
R32_ADC_IDATAR3	0x40012444	ADC injected data register 3	0x00000000
R32_ADC_IDATAR4	0x40012448	ADC injected data register 4	0x00000000
R32_ADC_RDATAR	0x4001244C	ADC regular data register	0x00000000

R32_ADC_DLYR	0x40012450	ADC delay register	0x00000000

8.3.1 ADC Status Register (ADC_STATR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											STRT	JSTRT	JEOC	EOC	AWD

Bit	Name	Access	Description	Reset value
[31:5]	Reserved	RO	Reserved	0
4	STRT	RW0	Rule channel transition start state. 1: Rule channel conversion has started. 0: Rule channel conversion is not started. This bit is set to 1 by hardware and cleared to 0 by software (write 1 is not valid).	0
3	JSTRT	RW0	Injection channel conversion start state. 1: Injection channel conversion has started. 0: Injection channel conversion has not started. This bit is set to 1 by hardware and cleared to 0 by software (write 1 is not valid).	0
2	JEOC	RW0	Injection into the end state of the channel group conversion. 1: Conversion complete. 0: The conversion is not completed. This bit is set to 1 by hardware (all injected channels are converted) and cleared to 0 by software (write 1 is invalid).	0
1	EOC	RW0	Conversion end state. 1: Conversion complete. 0: The conversion is not completed. This bit is set to 1 by hardware (end of rule or injection channel group conversion), cleared by software to 0 (write 1 is invalid) or when reading ADC_RDATAR.	0
0	AWD	RW0	Analog watchdog flag bit. 1: Occurrence of simulated watchdog events. 0: No simulated watchdog event occurred. This bit is set to 1 by hardware (conversion value is out of range of ADC_WDHTR and ADC_WDLTR registers) and cleared to 0 by software (write 1 is not valid).	0

8.3.2 ADC Control Register 1 (ADC_CTLR1)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								AWDEN	JAWDEN	Reserved					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DISCNUM[2:0]			JDISCEN	DISCEN	JAUTO	AWD	SCAN	JEOCIE	AWDIE	EOCI	AWDCH[4:0]				

				SGL				E	

Bit	Name	Access	Description	Reset value
[31:24]	Reserved	RO	Reserved	0
23	AWDEN	RW	Analog watchdog function enable bit on the rule channel. 1: Enable the analog watchdog on the rule channel. 0: Disable the analog watchdog on the rule channel.	0
22	JAWDEN	RW	Analog watchdog function enable bit on the injection channel. 1: Enable the analog watchdog on the injection channel. 0: Disable the analog watchdog on the injection channel.	0
[21:16]	Reserved	RO	Reserved	0
[15:13]	DISCNUM[2:0]	RW	Number of rule channels to be converted after external triggering in intermittent mode. 000: 1 channel. ... 111: 8 channels.	0
12	JDISCEN	RW	Inject the intermittent mode enable bit on the channel. 1: Enable intermittent mode on the injection channel. 0: Disable intermittent mode on the injection channel.	0
11	DISCEN	RW	Intermittent mode enable bit on rule channel. 1: Enable intermittent mode on the rule channel. 0: Disable intermittent mode on the rule channel.	0
10	JAUTO	RW	After the opening of the rule channel is completed, the injection channel group enable bit is automatically switched. 1: Enable automatic injection channel group switching. 0: Disable automatic injection channel group conversion. *Note: This mode requires disabling the external trigger function of the injection channel.*	0
9	AWDSGL	RW	In scan mode, use the analog watchdog enable bit on a single channel. 1: Use an analog watchdog on a single channel (AWDCH[4:0] selection). 0: Use analog watchdog on all channels.	0
8	SCAN	RW	Scan mode enable bit. 1: Enable scan mode (continuous conversion of all channels selected by ADC_IOFRx and ADC_RSQRx). 0: Disable scan mode.	0
7	JEOCIE	RW	Inject the channel group end-of-conversion interrupt enable bit. 1: Enable the injection of the channel group conversion completion interrupt (JEOC flag). 0: Disable the injection channel group conversion completion interrupt.	0
6	AWDIE	RW	Analog watchdog interrupt enable bit. 1: Enable the analog watchdog interrupt. 0: Disable the analog watchdog interrupt. *Note: In scan mode, this interrupt will abort the scan if it occurs.*	0
5	EOCIE	RW	End of conversion (rule or injection channel group) interrupt enable bit. 1: Enable the end-of-conversion interrupt (EOC flag). 0: Disable the end-of-conversion interrupt.	0
[4:0]	AWDCH[4:0]	RW	Analog watchdog channel selection bits. 00000: Analog input channel 0.	0

			00001: Analog input channel 1. ... 01111: Analog input channel 15.	

8.3.3 ADC Control Register 2 (ADC_CTLR2)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved	ADC_INJE_SWOFF	ADC_REGU_SWOFF	ADC_TRIG_SEL4	ADC_TRIG_SEL3	ADC_TRIG_SEL2	ADC_TRIG_SEL1	ADC_TRIG_SEL0	Reserved	SWSTART	JSWSTART	EXTTRIG	EXTSEL[2:0]			Reserved

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JEXTTRIG	JEXTSEL[2:0]			ALIGN	Reserved		DMA	Reserved				RSTCAL	CAL	CONT	ADON

Bit	Name	Access	Description	Reset value
31	Reserved	RO	Reserved	0
30	ADC_INJE_SWOFF	RW	Injection group channel conversion enable. 1: Injection group channel conversion is turned off; 0: Injection group channel conversion is enabled.	0
29	ADC_REGU_SWOFF	RW	Rule group channel conversion is enabled. 1: Rule group channel conversion is disabled; 0: Rule group channel conversion is enabled. 0	0
28	ADC_TRIG_SEL4	RW	The CC2 event of timer 2 is selected as an external trigger event for the rule group or the injection group. 1: As an external trigger event for the injection group. 0: As an external trigger event for the rule group.	0
27	ADC_TRIG_SEL3	RW	The CC1 event of timer 2 is selected as an external trigger event for the rule group or the injection group. 1: As an external trigger event for the injection group. 0: As an external trigger event for the rule group.	0
26	ADC_TRIG_SEL2	RW	The CC3 event of timer 1 is selected as an external trigger event for the rule group or the injection group. 1: As an external trigger event for the injection group. 0: As an external trigger event for the rule group.	0
25	ADC_TRIG_SEL1	RW	The CC2 event of timer 1 is selected as an external trigger event for the rule group or the injection group. 1: As an external trigger event for the injection group. 0: As an external trigger event for the rule group.	0
24	ADC_TRIG_SEL0	RW	The CC1 event of timer 1 is selected as an external trigger event for the rule group or the injection group. 1: As an external trigger event for the injection group. 0: As an external trigger event for the rule group.	0
23	Reserved	RO	Reserved	0
22	SWSTART	RW	To start a rule channel conversion, you need to set up a software trigger. 1: Start regular channel conversion; 0: Reset state. This bit is set by software and cleared by hardware after conversion begins.	0
21	JSWSTART	RW	To start an injection channel conversion, you need to set up a software trigger.	0

			1: Start injection channel conversion; 0: Reset state. This bit is set by software and cleared by hardware or cleared by software after conversion starts.	
20	EXTTRIG	RW	The external trigger conversion mode of the regular channel is enabled. 1: Use external events to start conversion; 0: Turn off external event start function.	0
[19:17]	EXTSEL[2:0]	RW	Selection of external trigger events that initiate rule channel conversion. 000: TRGO event of timer 1; 001: CC1 event of timer 1; 010: CC2 event of timer 1; 011: CC3 event of timer 1; 100: CC1 event of timer 2; 101: CC2 event of timer 2; 110: PA4/PA15 event; 111: SWSTART software trigger.	0
16	Reserved	RO	Reserved	0
15	JEXTTRIG	RW	Externally triggered conversion mode for the injected channel is enabled. 1: Use external events to start conversion; 0: Turn off external event start function.	0
[14:12]	JEXTSEL[2:0]	RW	Selection of external trigger events that initiate injection channel conversion. 000: Reserved; 001: CC1 event of timer 1; 010: CC2 event of timer 1; 011: CC3 event of timer 1; 100: CC1 event of timer 2; 101: CC2 event of timer 2; 110: PA4/PA15 event; 111: JSWSTART software trigger.	0
11	ALIGN	RW	Data alignment. 1: Left-aligned; 0: Right-aligned.	0
[10:9]	Reserved	RO	Reserved	0
8	DMA	RW	Direct Memory Access (DMA) mode enable. 1: Enable DMA mode; 0: Disable DMA mode.	0
[7:4]	Reserved	RO	Reserved	0
3	RSTCAL	RW	Reset calibration, this bit is set by software and cleared by hardware after reset is completed. 1: Initialize calibration register; 0: Calibration register has been initialized. *Note: If RSTCAL is set while a conversion is in progress, clearing the calibration register requires an additional cycle.*	0
2	CAL	RW	A/D calibration, this bit is set by software and cleared by hardware when calibration is completed. 1: Start calibration; 0: Calibration completed.	0
1	CONT	RW	Continuous conversion enable: 1: Continuous conversion mode; 0: Single conversion mode. If this bit is set, conversion will be continuous until the bit is cleared.	0
0	ADON	RW	On/off A/D converter When this bit is 0, writing 1 will wake up the ADC from	0

			power-down mode; when this bit is 1, writing 1 will start the conversion. 1: Turn on the ADC and start the conversion. 0: Turn off ADC conversion/calibration and enter power-down mode. *Note: A conversion is initiated when only ADON is changed in the register, and no new conversion is initiated if there are any other bits sent for change.*	

8.3.4 ADC Sample Time Configuration Register 1 (ADC_SAMPTR1)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															SMP15[1]

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15[0]	Reserved	SMP14[1:0]		Reserved	SMP13[1:0]		Reserved	SMP12[1:0]		Reserved	SMP11[1:0]		Reserved	SMP10[1:0]	

Bit	Name	Access	Description	Reset value
[31:17] 14 11 8 5 2	Reserved	RO	Reserved	0
[16:15] [13:12] [10:9] [7:6] [4:3] [1:0]	SMPx[1:0]	RW	SMPx[1:0], x=10~15: Sampling time configuration of channel x. 00: 3 cycles; 01: 13 cycles; 10: 37 cycles; 11: 49 cycles; These bits are used to select the sampling time for each channel independently, and the channel configuration value must remain constant during the sampling cycle.	0

8.3.5 ADC Sample Time Configuration Register 2 (ADC_SAMPTR2)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			SMP9[1:0]		Reserved	SMP8[1:0]		Reserved	SMP7[1:0]		Reserved	SMP6[1:0]		Reserved	SMP5[1]

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5[0]	Reserved	SMP4[1:0]		Reserved	SMP3[1:0]		Reserved	SMP2[1:0]		Reserved	SMP1[1:0]		Reserved	SMP0[1:0]	

Bit	Name	Access	Description	Reset value
[31:29] 26 23 20 17	Reserved	RO	Reserved	0

14 11 8 5 2				
[28:27] [25:24] [22:21] [19:18] [16:15] [13:12] [10:9] [7:6] [4:3] [1:0]	SMPx[1:0]	RW	SMPx[1:0], x = 0~9: Sampling time configuration of channel x. 00: 3 cycles; 01: 13 cycles; 10: 37 cycles; 11: 49 cycles; These bits are used to select the sampling time for each channel independently, and the channel configuration value must remain constant during the sampling cycle.	

8.3.6 ADC Injected Channel Data Offset Register x (ADC_IOFRx) (x=1/2/3/4)

Offset address: 0x14 + (x-1)*4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						JOFFSETx[9:0]									

Bit	Name	Access	Description	Reset value
[31:10]	Reserved	RO	Reserved	0
[9:0]	JOFFSETx[9:0]	RW	The data offset value of the injected channel x. When converting the injected channels, this value defines the value used to subtract from the original conversion data. The result of the conversion can be read out in the ADC_IDATARx register.	0

8.3.7 ADC Watchdog High Threshold Register (ADC_WDHTR)

Offset address: 0x24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						HT[9:0]									

Bit	Name	Access	Description	Reset value
[31:10]	Reserved	RO	Reserved	0
[9:0]	HT[9:0]	RW	Analog watchdog high threshold setting value.	3FFh

Note: You can change the values of WDHTR and WDLTR during the conversion process, but they will take effect at the next conversion.

8.3.8 ADC Watchdog Low Threshold Register (ADC_WDLTR)

Offset address: 0x28

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						LT[9:0]									

Bit	Name	Access	Description	Reset value
[31:10]	Reserved	RO	Reserved	0
[9:0]	LT[9:0]	RW	Analog watchdog low threshold setting value.	0

Note: You can change the values of WDHTR and WDLTR during the conversion process, but they will take effect at the next conversion.

8.3.9 ADC Regular Sequence Register 1(ADC_RSQR1)

Offset address: 0x2C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								L[3:0]				SQ16[4:1]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ16[0]	SQ15[4:0]					SQ14[4:0]					SQ13[4:0]				

Bit	Name	Access	Description	Reset value
[31:24]	Reserved	RO	Reserved	0
[23:20]	L[3:0]	RW	Number of channels to be converted in a regular channel conversion sequence. 0000-1111: 1-16 conversions.	0
[19:15]	SQ16[4:0]	RW	The number of the 16th conversion channel in the rule sequence (0-15).	0
[14:10]	SQ15[4:0]	RW	The number of the 15th conversion channel in the rule sequence (0-15).	0
[9:5]	SQ14[4:0]	RW	The number of the 14th conversion channel in the rule sequence (0-15).	0
[4:0]	SQ13[4:0]	RW	The number of the 13th conversion channel in the rule sequence (0-15).	0

8.3.10 ADC Regular Sequence Register 2 (ADC_RSQR2)

Offset address: 0x30

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		SQ12[4:0]						SQ11[4:0]				SQ10[4:1]			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ10[0]	SQ9[4:0]					SQ8[4:0]					SQ7[4:0]				

Bit	Name	Access	Description	Reset value
[31:30]	Reserved	RO	Reserved	0
[29:25]	SQ12[4:0]	RW	The number of the 12th conversion channel in the rule sequence (0-15).	0
[24:20]	SQ11[4:0]	RW	The number of the 11th conversion channel in the rule sequence (0-15).	0
[19:15]	SQ10[4:0]	RW	The number of the 10th conversion channel in the rule sequence (0-15).	0

[14:10]	SQ9[4:0]	RW	The number of the 9th conversion channel in the rule sequence (0-15).	0
[9:5]	SQ8[4:0]	RW	The number of the 8th conversion channel in the rule sequence (0-15).	0
[4:0]	SQ7[4:0]	RW	The number of the 7th conversion channel in the rule sequence (0-15).	0

8.3.11 ADC Regular Sequence Register 3 (ADC_RSQR3)

Offset address: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		SQ6[4:0]					SQ5[4:0]					SQ4[4:1]			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ4[0]	SQ3[4:0]					SQ2[4:0]					SQ1[4:0]				

Bit	Name	Access	Description	Reset value
[31:30]	Reserved	RO	Reserved	0
[29:25]	SQ6[4:0]	RW	The number of the 6th conversion channel in the rule sequence (0-15).	0
[24:20]	SQ5[4:0]	RW	The number of the 5th conversion channel in the rule sequence (0-15).	0
[19:15]	SQ4[4:0]	RW	The number of the 4th conversion channel in the rule sequence (0-15).	0
[14:10]	SQ3[4:0]	RW	The number of the 3th conversion channel in the rule sequence (0-15).	0
[9:5]	SQ2[4:0]	RW	The number of the 2th conversion channel in the rule sequence (0-15).	0
[4:0]	SQ1[4:0]	RW	The number of the 1th conversion channel in the rule sequence (0-15).	0

8.3.12 ADC Injected Sequence Register (ADC_ISQR)

Offset address: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved										JL[1:0]		JSQ4[4:1]			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ4[0]	JSQ3[4:0]					JSQ2[4:0]					JSQ1[4:0]				

Bit	Name	Access	Description	Reset value
[31:22]	Reserved	RO	Reserved	0
[21:20]	JL[1:0]	RW	Inject the number of channels to be converted in the channel conversion sequence. 00-11: 1-4 conversions.	0
[19:15]	JSQ4[4:0]	RW	The number of the 4th conversion channel in the injection sequence (0-15). *Note: Software writes and assigns the channel number (0-15) as the 4th in the sequence to be converted.*	0
[14:10]	JSQ3[4:0]	RW	The number of the 3th conversion channel in the injection sequence (0-15).	0
[9:5]	JSQ2[4:0]	RW	The number of the 2th conversion channel in the injection sequence (0-15).	0

[4:0]	JSQ1[4:0]	RW	The number of the 1th conversion channel in the injection sequence (0-15).	0

Note: Unlike the regular conversion sequence, if the length of JL[1:0] is less than 4, the sequence order of conversion starts from (4 - JL).

For example, when JL[1:0]=3 (4 injected transitions in the sequencer), the ADC will convert channels in the following order: JSQ1[4:0], JSQ2[4:0], JSQ3[4:0], and JSQ4[4:0];

When JL[1:0]=2 (3 injected transitions in the sequencer), the ADC will convert the channels in the following order: JSQ2[4:0], JSQ3[4:0] and JSQ4[4:0];

When JL[1:0]=1 (2 injected conversions in the sequencer), the ADC converts the channels in the following order: first JSQ3[4:0], then JSQ4[4:0];

When JL[1:0] = 0 (1 injection conversion in the sequencer), the ADC will convert only the JSQ4[4:0] channels.

If ADCx_ISQR[21:0]=10 00111 00011 00111 00010, the ADC will convert channels in the following order: JSQ2[4:0], JSQ3[4:0], and JSQ4[4:0], indicating that the scan conversions are performed in the following channel order: 7, 3, 7.

8.3.13 ADC Injected Data Register (ADC_IDATARx) (x=1/2/3/4)

Offset address: 0x3C + (x-1)*4

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JDATA[15:0]															

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:0]	JDATA[15:0]	RO	Injection of channel conversion data (data left-aligned or right-aligned).	0

8.3.14 ADC Regular Data Register (ADC_RDATAR)

Offset address: 0x4C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:0]	DATA[15:0]	RO	Regular channel conversion data (data left-aligned or right-aligned).	0

8.3.15 ADC Delay Register (ADC_DLYR)

Offset address: 0x50

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						DLYSRC	DLYVLU[8:0]								

Bit	Name	Access	Description	Reset value
[31:10]	Reserved	RO	Reserved	0
9	DLYSRC	RW	External trigger source delay selection. 0: Rule channel external trigger delay; 1: Injection channel external trigger delay.	0
[8:0]	DLYVLU[8:0]	RW	External trigger delay data, delay time configuration, unit: ADC clock cycle.	0

Chapter 9 Advanced-control Timer (ADTM)

The Advanced-control timer Module contains one powerful 16-bit auto-reload timer TIM1, which can be used to measure pulse width or generate pulses, PWM waves, etc. It is used in motor control, power supply, etc.

9.1 Main Features

The main features of the advanced-control timer TIM1 include.

- 16-bit auto-reload counter supporting incremental counting mode, decremental counting mode and incremental and decremental counting mode.
- 16-bit prescaler with dynamically adjustable crossover coefficients from 1 to 65536.
- Support 3 independent comparison capture channels.
- Each comparison capture channel supports multiple operating modes, such as: input capture, output comparison, PWM generation and single pulse output.
- Support complementary outputs with programmable dead time;
- Support external signals to control the timer.
- Support updating the timer after a defined period using a repeat counter.
- Support resetting the timer or placing it in the OK state using the brake signal.
- Support the use of DMA in multiple modes.
- Support incremental encoders.
- Support cascading and synchronization between timers.

9.2 Principle and Structure

This section deals with the internal construction of advanced-control timers.

9.2.1 Overview

As shown in Figure 9-1, the structure of the advanced-control timer can be roughly divided into 3 parts, namely the input clock part, the core counter part and the compare capture channel part.

The advanced-control timer can be clocked from the HB bus clock (CK_INT), from an external clock input pin (TIMx_ETR), from other timers with clock output (ITRx), or from the input of the compare capture channel (TIMx_CHx). These input clock signals become the CK_PSC clock after various set filtering and dividing operations and are output to the core counter section. In addition, these complex clock sources can also be output as TRGO to other peripherals such as timers and ADCs.

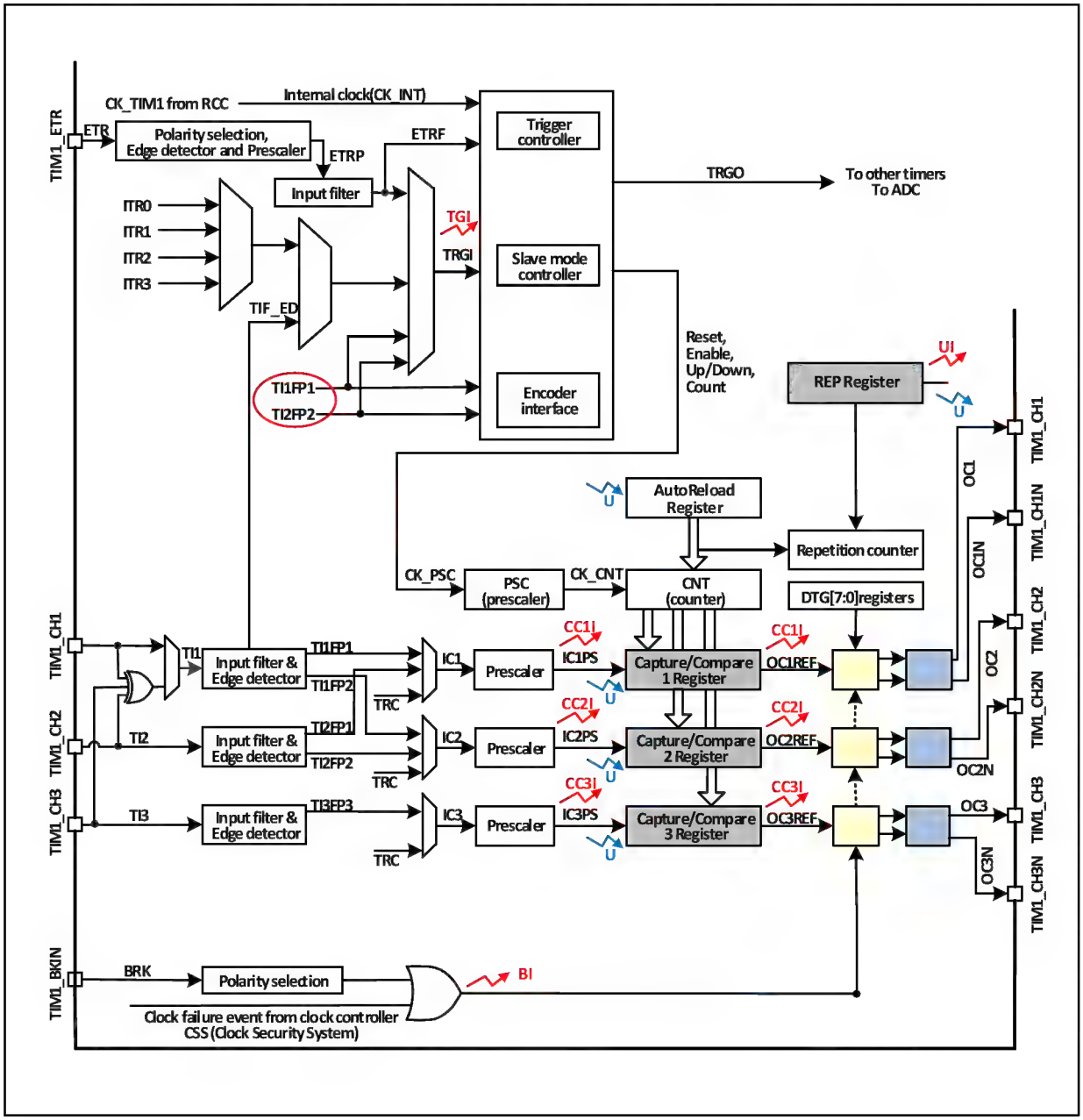
The core of the advanced timer is a 16-bit counter (CNT). After CK_PSC is divided by the prescaler (PSC), it becomes CK_CNT and is output to CNT. CNT supports up counting mode, down counting mode and up and down counting mode, and has an automatic reload value register (ATRLR) in each count After the cycle ends, the CNT is reloaded with the initial value. There is also an auxiliary counter that counts the number of times ATRLR reloads the initial value for CNT. When the number reaches the number set in the repetition count value register (RPTCR), a specific event can be generated.

The core of the advanced timer is a 16-bit counter (CNT). After CK_PSC is divided by the prescaler (PSC), it becomes CK_CNT and is output to CNT. CNT supports up counting mode, down counting mode and up and down counting mode, and has an automatic reload value register (ATRLR) in each count After the cycle ends, the CNT is reloaded with the initial value. There is also an auxiliary counter that counts the number of times ATRLR reloads

the initial value for CNT. When the number reaches the number set in the repetition count value register (RPTCR), a specific event can be generated.

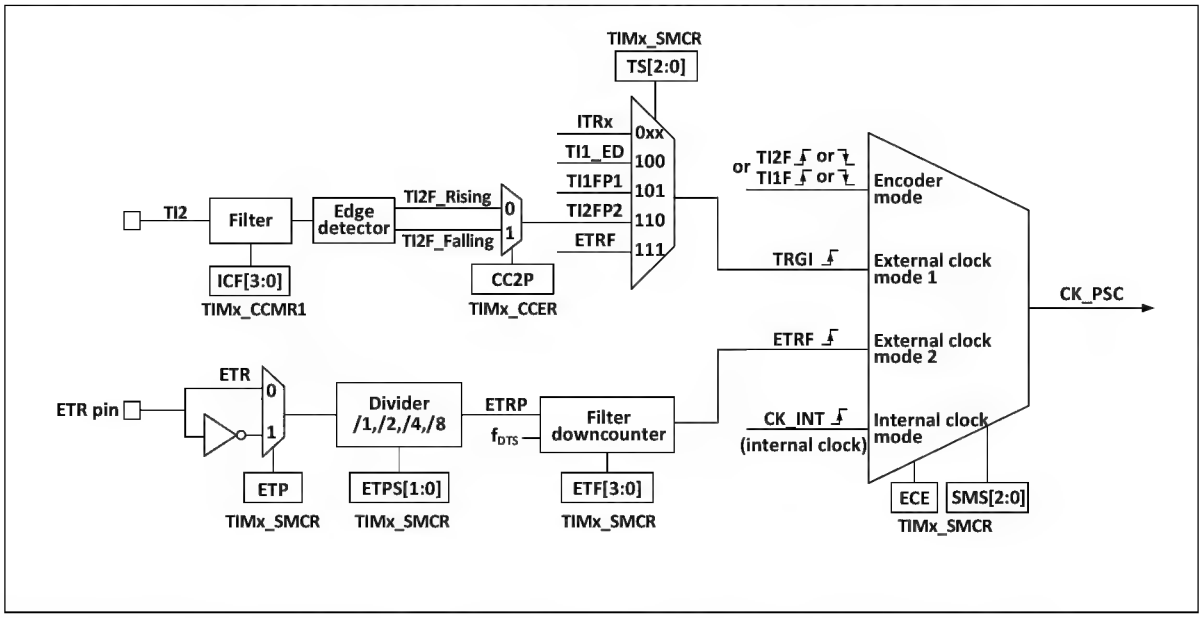
The advanced timer has three sets of compare capture channels. Each set of compare capture channels can input pulses from its own pins and can also output waveforms to the pins. That is, the compare capture channels support input and output modes. The input of each channel of the compare capture register supports operations such as filtering, frequency division, and edge detection, supports mutual triggering between channels, and can also provide a clock for the core counter CNT. Each compare capture channel has a set of compare capture registers (CHxCVR) that support comparison with the main counter (CNT) and output pulses.

Figure 9-1 Block diagram of advanced-control timer structure



9.2.2 Clock Input

Figure 9-2 Block diagram of CK_PSC source for advanced-control timer



The advanced-control timer CK_PSC has many clock sources and can be divided into 4 categories.

1) Route of external clock pin (ETR) input clock: ETR → ETRP → ETRF.
2) Internal HB clock input route: CK_INT.
3) Route from the comparison capture channel pin (TIMx_CHx): TIMx_CHx → TIx → TIxFPx, this route is also used in encoder mode.
4) Inputs from other internal timers: ITRx.

The actual operation can be divided into 4 categories by determining the choice of input pulse for the SMS of the CK_PSC source.

1) Selection of the internal clock source (CK_INT).
2) External clock source mode 1.
3) External clock source mode 2.
4) Encoder mode.

All 4 clock source sources mentioned above can be selected by these 4 operations.

9.2.2.1 Internal Clock Source (CK_INT)

If the SMS field is held at 000b to start the advanced-control timer, then it is the internal clock source (CK_INT) that is selected as the clock. At this point CK_INT is CK_PSC.

9.2.2.2 External Clock Source Mode1

When the SMS domain is set to 111b, external clock source mode 1 is enabled. When external clock source 1 is enabled, TRGI is selected as the source of CK_PSC. it is worth noting that the source of TRGI also needs to be selected by configuring the TS domain. the TS domain can select the following types of pulses as clock sources.

1) Internal trigger (ITRx, x is 0,1,2,3).
2) Comparison of the signal after capturing channel 1 through the edge detector (TI1F_ED).
3) Comparison of signals TI1FP1, TI2FP2 of the capture channel.
4) The signal ETRF from the external clock pin input.

9.2.2.3 External Clock Source Mode2

Use external trigger mode 2 to count on every rising or falling edge of the external clock pin input. When the ECE position is set, the external clock source mode 2 is used. when using the external clock source mode 2, ETRF is selected as CK_PSC. the ETR pin becomes ETRP after passing through the optional inverter (ETP), divider (ETPS), and then ETRF after passing through the filter (ETF).

With the ECE position bit and the SMS set to 111b, this is equivalent to the TS selecting ETRF as an input.

9.2.2.4 Encoder Mode

Setting the SMS to 001b, 010b, 011b will enable the encoder mode. Enabling encoder mode allows you to select a specific level in TI1FP1 and TI2FP2 to signal the output with another jump edge as the signal. This mode is used when an external encoder is used. Refer to Section 9.3.9 for specific functions.

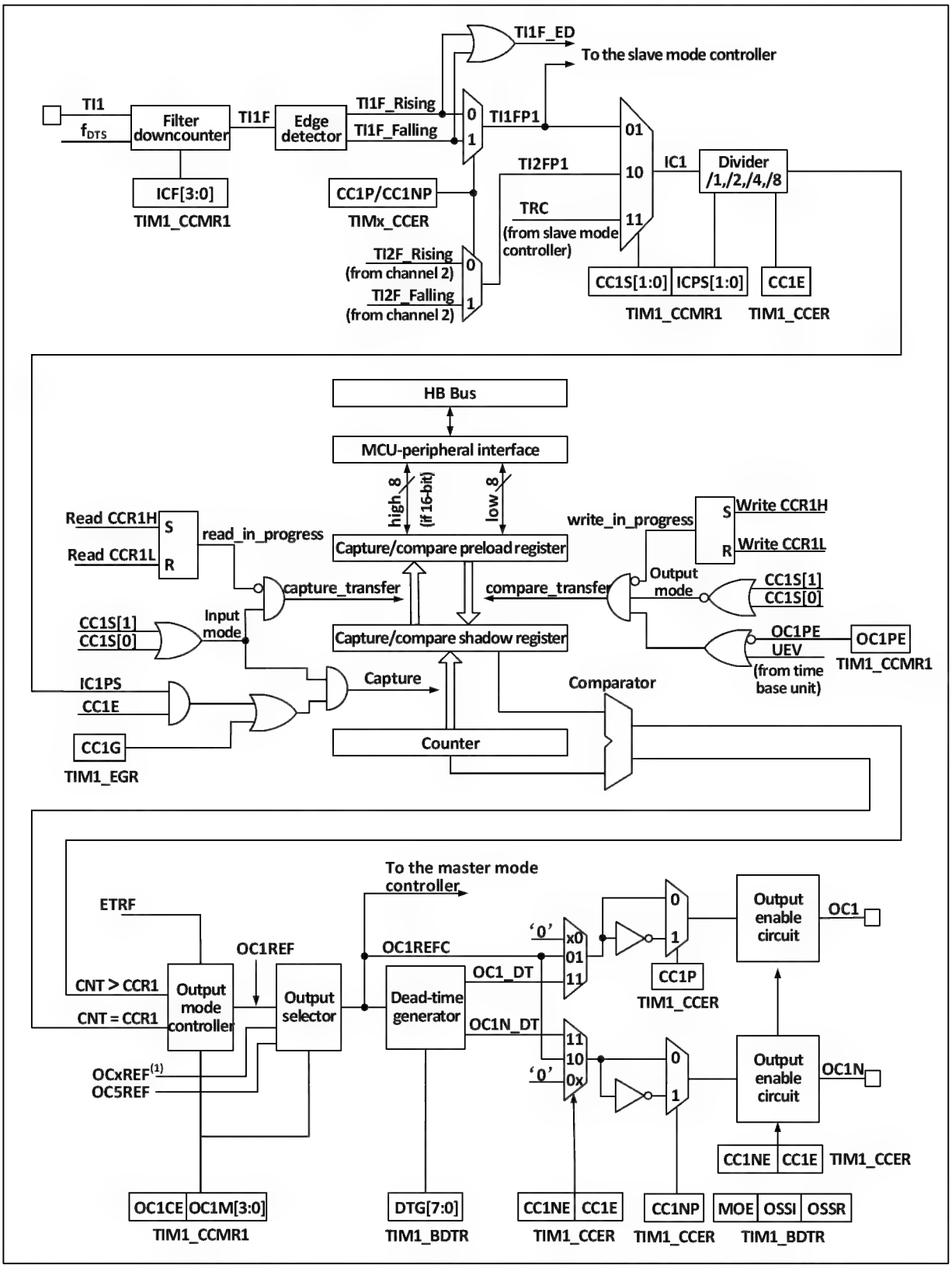
9.2.3 Counters and Peripherals

CK_PSC is input to the prescaler (PSC) for frequency division. PSC is 16 bits, and the actual frequency division coefficient is equivalent to the value of R16_TIM1_PSC + 1. CK_PSC will become CK_INT after passing through PSC. Changing the value of R16_TIM1_PSC will not take effect in real time, but will be updated to PSC after the update event. Update events include clearing and resetting the UG bit. The core of the timer is a 16-bit counter (CNT). CK_CNT will eventually be input to CNT. CNT supports up counting mode, down counting mode and up and down counting mode, and has an automatic reload value register (ATRLR) in each count. After the cycle ends, the CNT is reloaded with initial values. There is also an auxiliary counter that records the number of times ATRLR reloads the initial value for CNT. When the number of times set in the repetition count value register (RPTCR) is reached, a specific event can be generated.

9.2.4 Compare/Capture Channels and Peripherals

The compare capture channel is the main component for the timer to implement complex functions. Its core is the compare capture register, supplemented by digital filtering of the peripheral input part, frequency division and inter-channel multiplexing, and the comparator and output control of the output part.

Figure 9-3 Block diagram of the structure of the compare/capture channel



The structure block diagram of the compare/capture channel is shown in Figure 9-3. The signal is input from the channel x pin and optionally made as TIx (the source of TI1 can be more than just CH1, see the structure block diagram of timer 9-1), TI1 is passed through the filter (ICF[3:0]) to generate TI1F, and then divided into TI1F_Rising and TI1F_Falling through the edge detector, these two signals are selected (CC1P) to generate TI1FP1, TI1FP1 and

TI2FP1 from channel 2 are sent together to CC1S to select to become IC1, which is sent to the compare capture register after ICPS dividing.

The compare capture register consists of a preload register and a shadow register, and the read/write process operates only on the preload register. In capture mode, the capture occurs on the shadow register and is then copied to the preload register; in compare mode, the contents of the preload register are copied to the shadow register, and then the contents of the shadow register are compared to the core counter (CNT).

9.3 Function and Implementation

The implementation of advanced timer complex functions is achieved by operating the timer's comparison capture channel, clock input circuit, counter and peripheral parts. The clock input to the timer can come from multiple clock sources including the input to the compare capture channel. The operation of comparing the capture channel and clock source selection directly determines its function. The compare capture channel is bidirectional and can work in input and output modes.

9.3.1 Input Capture Mode

Input capture mode is one of the basic functions of the timer. The principle of the input capture mode is that when a certain edge on the ICxPS signal is detected, a capture event occurs, and the current value of the counter will be latched into the comparison capture register (R16_TIM1_CHCTLRx). When a capture event occurs, CCxIF (in R16_TIM1_INTFR) is set, and if interrupts or DMA are enabled, corresponding interrupts or DMA will also be generated. If CCxIF is already set when a capture event occurs, the CCxOF bit will be set. CCxIF can be cleared by software or by hardware by reading the compare capture register. CCxOF is cleared by software.

Take an example of channel 1 to illustrate the steps of using input capture mode, as follows:

1) Configure the CCxS domain and select the source of the ICx signal. For example, set it to 10b and select TI1FP1 as the source of IC1 instead of using the default settings. The CCxS domain defaults to the comparison capture module as the output channel;
2) Configure the ICxF domain and set the digital filter for the TI signal. The digital filter will sample a certain frequency at a certain frequency, and then output a transition. This sampling frequency and number of times are determined by ICxF;
3) Configure the CCxP bit, set the polarity of TIxFPx. For example, keep the CC1P bit low and select rising edge transition;
4) Configure the ICxPS domain, set ICx signal to be the prescaler factors between ICxPS. Such as keep ICxPS as 00b, no prescaler;
5) Configure the CCxE bit allows capturing the value of the core counter (CNT) into the compare capture register. Set CC1E bit;
6) Configure the CCxIE and CCxDE bits as needed to decide whether to enable interrupts or DMA.

At this point, the comparison capture channel configuration has been completed.

When TI1 inputs a captured pulse, the value of the core counter (CNT) will be recorded in the compare capture register, and CC1IF will be set. When CC1IF has been set before, the CCIOF bit will also be set. If CC1IE is set, an interrupt will be generated; if CC1DE is set, a DMA request will be generated. An input capture event can be generated by software by writing to the event generation register (TIM1_SWEVGR).

9.3.2 Compare Output Mode

Compare output mode is one of the basic functions of a timer. The principle of the compare output mode is to output a specific change or waveform when the value of the core counter (CNT) is consistent with the value of the compare

capture register. The OCxM field (in R16_TIM1_CHCTLRx) and the CCxP bit (in R16_TIM1_CCER) determine whether the output is a definite high and low level or a level flip. When a comparison consistency event occurs, the CCxIF bit will also be set. If the CCxIE bit is set in advance, an interrupt will be generated; if the CCxDE bit is set in advance, a DMA request will be generated.

The steps to configure comparison output mode are as follows:

1) Configure the clock source and auto-reload value of the core counter (CNT);
2) Set the count value to be compared to the comparison capture register (R16_TIM1_CHxCVR);
3) If an interrupt needs to be generated, set the CCxIE bit;
4) Keep OCxPE at 0 to disable the preload register of the compare register;
5) Set the output mode, set the OCxM field and CCxP bit;
6) Enable the output and set the CCxE bit;
7) Set the CEN bit to start the timer.

9.3.3 Forced Output Mode

The output mode of the timer's compare capture channel can be forced to output a determined level by software without relying on the comparison of the shadow register of the compare capture register and the core counter.

This is done by setting OCxM to 100b, which forces OCxREF to be set low, or by setting OCxM to 101b, which forces OCxREF to be set high.

It should be noted that when OCxM is forced to 100b or 101b, the comparison process of the internal core counter and the compare capture register is still in progress, the corresponding flag bit is still set, and interrupts and DMA requests are still generated.

9.3.4 PWM Input Mode

PWM input mode is used to measure the duty cycle and frequency of PWM. It is a special case of input capture mode. The operation is the same as input capture mode except for the following differences: PWM occupies two compare capture channels, and the input polarity of the two channels is set to opposite, one of the signals is set as the trigger input, and SMS is set in reset mode.

For example, to measure the period and frequency of the PWM wave input from TI1, you need to perform the following operations:

1) Set TI1 (TI1FP1) to the input of the IC1 signal. Set CC1S to 01b;
2) Set TI1FP1 to rising edge active. Hold CC1P at 0;
3) Set TI1 (TI1FP2) to the input of the IC2 signal. Set CC2S to 10b;
4) Select TI1FP2 to be falling edge active. Set CC2P to 1;
5) Select TI1FP1 for the source of the clock source. set TS to 101b;
6) Set SMS to reset mode, i.e. 100b;
7) Enable input capture. CC1E and CC2E are set;

In this way, the value of comparison capture register 1 is the period of PWM, and the value of comparison capture register 2 is its duty cycle.

Note: Since only TI1FP1 and TI2FP2 are connected to the slave mode controller, only TIM1_CH1/TIM1_CH2 can be used in the PWM input mode.

9.3.5 PWM Output Mode

PWM output mode is one of the basic functions of the timer. The most common method of PWM output mode is to use the reload value to determine the PWM frequency and the capture compare register to determine the duty cycle.

Set the OCxM field to 110b or 111b to use PWM mode 1 or mode 2, set the OCxPE bit to enable the preload register, and finally set the ARPE bit to enable automatic reloading of the preload register. Since the value of the preload register can be sent to the shadow register when an update event occurs, the UG bit needs to be set to initialize all registers before the core counter starts counting. In PWM mode, the core counter and compare capture register are constantly comparing, and the timer can output edge-aligned or center-aligned PWM signals according to the CMS bit.

- Edge aligned

When edge aligned is used, the core counter counts up or down. In the case of PWM mode 1, when the value of the core counter is greater than the compare capture register, OCxREF is high; when the value of the core counter is less than the compare capture register (such as the core counter grows to the value of R16_TIM1_ATRLR and returns to all 0), OCxREF is low.

- Center aligned

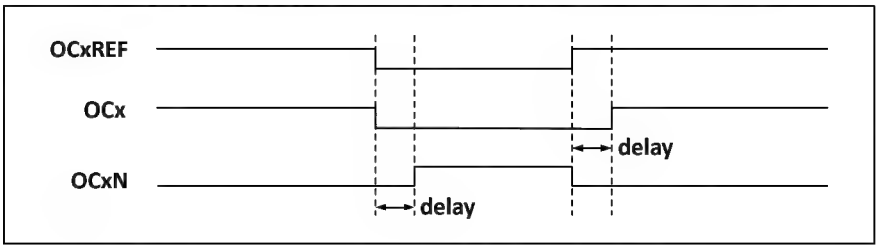
When using the center aligned mode, the core counter runs in an alternating up-counting and down-counting mode, and OCxREF performs rising and falling transitions when the values of the core counter and the compare capture register are consistent. However, the timing of setting the comparison flag is different in the three center aligned modes. When using center aligned mode, it is best to generate a software update flag (set the UG bit) before starting the core counter.

9.3.6 Complementary Outputs and Dead-time Insertion

The compare capture channel generally has two output pins, which can output two complementary signals (OCx and OCxN). OCx and OCxN can independently set the polarity through the CCxP and CCxNP bits, and independently set the output enable through CCxE and CCxNE. Dead-time and other controls are performed through MOE, OIS, OISN, OSSI, and OSSR bits. Enabling both the OCx and OCxN outputs will insert dead-time, with each channel having a 10-bit dead-time generator. If there is a brake circuit, the MOE bit must also be set. OCx and OCxN are generated by the association of OCxREF. If OCx and OCxN are both high and active, then OCx is the same as OCxREF, except that the rising edge of OCx is equivalent to OCxREF with a delay. OCxN is opposite to OCxREF. Its rising edge is relative to the falling edge of the reference signal. There will be a delay, and if the delay is greater than the effective output width, the corresponding pulse will not be generated.

Figure 9-4 shows the relationship between OCx, OCxN and OCxREF, and shows the dead-time.

Figure 9-4 Complementary outputs and dead-time insertion



9.3.7 Brake Signal

When the brake signal is generated, the output enable signal and invalid level will be modified according to the MOE, OIS, OISN, OSSI and OSSR bits. But OCx and OCxN will not be at active levels at any time. The brake event source can come from the brake input pin.

After the system is reset, the braking function is disabled by default (MOE bit is low). Setting the BKE bit can enable the braking function. The polarity of the input braking signal can be set by setting BKP. The BKE and BKP signals can be written at the same time. There will be an HB clock delay before actual writing, so you need to wait

for an HB cycle to correctly read the written value.

When the selected level appears on the brake pin, the system will produce the following actions:

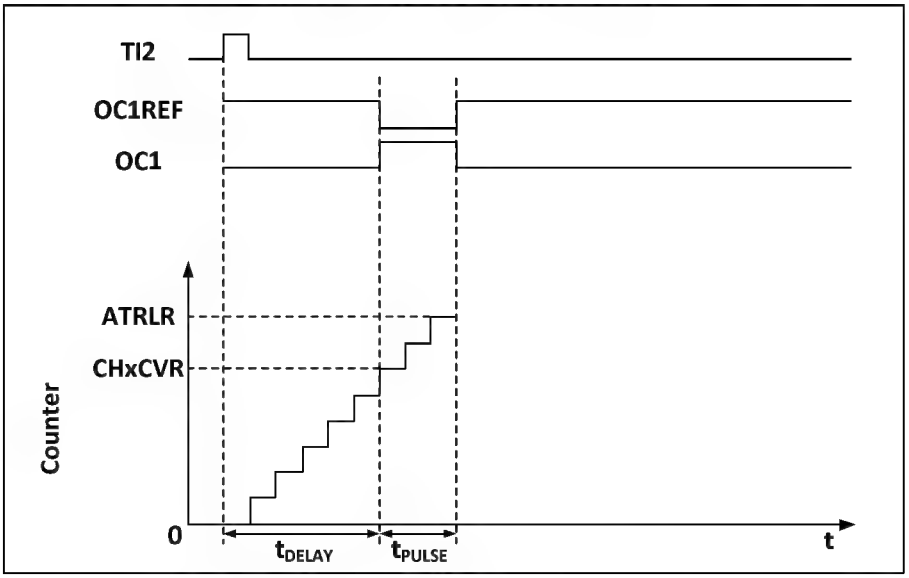
1) The MOE bit is cleared asynchronously, and the output is set to the inactive state, idle state, or reset state according to the setting of the SOOI bit;
2) After MOE is cleared, each output channel outputs the level determined by OISx;
3) When using complementary outputs: the outputs are put into an inactive state, depending on the polarity;
4) If BIE is set, an interrupt will be generated when BIF is set; if the BDE bit is set, a DMA request will be generated;
5) If AOE is set, the MOE bit is automatically set on the next update event UEV.

9.3.8 One-pulse Mode

One-pulse mode (OPM) can be used to allow the microcontroller to respond to a specific event by generating a pulse after a delay. The delay and pulse width are programmable. Setting the OPM bit can cause the core counter to stop when the next update event UEV is generated (the counter rolls over to 0).

As shown in Figure 9-5, it is necessary to detect a rising edge on the TI2 input pin. After delaying Tdelay, a positive pulse of length Tpulse is generated on OC1:

Figure 9-5 Generation of one pulse mode



1) Set TI2 as trigger. Set CC2S field to 01b to map TI2FP2 to TI2; set CC2P bit to 0b to set TI2FP2 as rising edge detection; set TS field to 110b to set TI2FP2 as trigger source; and set SMS field to 110b for TI2FP2 to be used to start the counter;
2) Tdelay is determined by the value of the compare capture register, and Tpulse is determined by the value of the auto-reload value register and the value of the compare capture register.

9.3.9 Encoder Interface Mode

Encoder interface mode is a typical application of the timer. It can be used to access the bi-phase output of the encoder. The counting direction of the core counter is synchronized with the rotation axis direction of the encoder. Each time the encoder outputs a pulse, the core counter will increase by one or minus one. The steps to use the encoder are: set the SMS field to 001b (counting only on TI2 edges), 010b (counting only on TI1 edges) or 011b (counting on both TI1 and TI2 edges), and connect the encoder to compare capture channel 1, 2 input terminal, set a value for the reload value register, this value can be set larger. In encoder mode, the timer's internal compare

capture register, prescaler, repetition count register, etc. all work normally. The following table shows the relationship between counting direction and encoder signal.

Table 9-1 Relationship between counting direction and encoder signal of timer encoder mode

Counting effective edges	The level of relative signals	TI1FP1 signal edge		TI2FP2 signal	
		Rising edge	Falling edge	Rising edge	Falling edge
Counting at TI1 edge only	high	Downward counting	Upward counting	No count	
	low	Upward counting	Downward counting		
Counting at TI2 edge only	high	No count		Upward counting	Downward counting
	low			Downward counting	Upward counting
Double edge counting at TI1 and TI2	high	Downward counting	Upward counting	Upward counting	Downward counting
	low	Upward counting	Downward counting	Downward counting	Upward counting

9.3.10 Timer Synchronization Mode

The timer can output clock pulses (TRGO) and can also receive input from other timers (ITRx). The sources of ITRx of different timers (TRGO of other timers) are different. The internal trigger connection of the timer is shown in Table 9-2.

Table 9-2 TIMx internal trigger connections

Slave	ITR0(TS=000)	ITR1(TS=001)	ITR2(TS=010)	ITR3(TS=011)
TIM2	TIM1			

9.3.11 Debug Mode

When the system enters debug mode, the timer continues to run or stops according to the settings of the DBG module.

9.4 Register Description

Table 9-3 TIM1-related registers list

Name	Access address	Description	Reset value
R16_TIM1_CTLR1	0x40012C00	Control register 1	0x0000
R16_TIM1_CTLR2	0x40012C04	Control register 2	0x0000
R16_TIM1_SMCFGR	0x40012C08	Slave mode control register	0x0000
R16_TIM1_DMAINTENR	0x40012C0C	DMA/interrupt enable register	0x0000
R16_TIM1_INTFR	0x40012C10	Interrupt status register	0x0000
R16_TIM1_SWEVGR	0x40012C14	Event generation register	0x0000
R16_TIM1_CHCTLR1	0x40012C18	Compare/capture control register 1	0x0000
R16_TIM1_CHCTLR2	0x40012C1C	Compare/capture control register 2	0x0000
R16_TIM1_CCER	0x40012C20	Compare/capture enable register	0x0000
R16_TIM1_CNT	0x40012C24	Counters	0x0000
R16_TIM1_PSC	0x40012C28	Counting clock prescaler	0x0000
R16_TIM1_ATRLR	0x40012C2C	Auto-reload value register	0xFFFF
R16_TIM1_RPTCR	0x40012C30	Repeat count Register	0x0000

R32_TIM1_CH1CVR	0x40012C34	Compare/capture register 1	0x00000000
R32_TIM1_CH2CVR	0x40012C38	Compare/capture register 2	0x00000000
R32_TIM1_CH3CVR	0x40012C3C	Compare/capture register 3	0x00000000
R16_TIM1_BDTR	0x40012C44	Break and dead-time register	0x0000
R16_TIM1_DMACFGR	0x40012C48	DMA control register	0x0000
R16_TIM1_DMAADR	0x40012C4C	DMA address register for continuous mode	0x0000

9.4.1 Control Register 1 (TIM1_CTLR1)

Offset address: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CAPLVL	CAPOV	Reserved				CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN

Bit	Name	Access	Description	Reset value
15	CAPLVL	RW	In double-edge capture mode, the capture level indication is enabled. 0: Disable the indication function 1: Enable the indication function. *Note: When enabled, CHxCVR[16] indicates the level corresponding to the capture value.*	0
14	CAPOV	RW	Capture value mode configuration. 0: The capture value is the value of the actual counter 1: The CHxCVR value is 0xFFFF when a counter overflow is generated before capture.	0
[13:10]	Reserved	RO	Reserved	0
[9:8]	CKD[1:0]	RW	These 2 bits define the division ratio between the timer clock (CK_INT) frequency, the dead time and the sampling clock used by the dead time generator and the digital filter (ETR,TIx). 00: Tdts=Tck_int 01: Tdts = 2 × Tck_int 10: Tdts = 4 × Tck_int 11: Reserved.	0
7	ARPE	RW	Auto-reload preload enable bit. 1: Enable Auto-reload Value Register (ATRLR). 0: Disable Auto-reload Value Register (ATRLR).	0
[6:5]	CMS[1:0]	RW	Center aligned mode selection. 00: Edge-aligned mode. The counter counts up or down based on the direction bit (DIR). 01: Center-aligned mode 1. The counter counts up and down alternately. The output compare interrupt flag bit of the channel configured as output (CCxS=00 in the CHCTLRx register) is set only when the counter counts down. 10: Center-aligned mode 2. The counter counts up and down alternately. The output compare interrupt flag bit of the channel configured as output (CCxS=00 in the CHCTLRx register) is set only when the counter counts up. 11: Center-aligned mode 3. The counter counts up and down alternately. The output compare interrupt flag bit of the channel configured as output (CCxS=00 in the CHCTLRx register) is set when the counter counts both up and down.	0

			Note: When the counter is enabled (CEN=1), the transition from edge-aligned mode to center-aligned mode is not allowed.	
4	DIR	RW	Counting direction. 0: The counting mode of the counter is up counting; 1: The counting mode of the counter is down counting. *Note: This bit is not valid when the counter is configured in center-aligned mode or encoder interface mode.*	0
3	OPM	RW	One-pulse mode. 1: The counter stops (clear the CEN bit) when the next update event occurs. 0: The counter does not stop when the next update event occurs.	0
2	URS	RW	Update request source, by which the software selects the source of the UEV event. 1: If an update interrupt or DMA request is enabled, only an update interrupt or DMA request is generated if the counter overflows/underflows. 0: If an update interrupt or DMA request is enabled, an update interrupt or DMA request is generated by any of the following events. -Counter overflow/underflow -Setting the UG position -Updates generated by the slave mode controller	0
1	UDIS	RW	Disable updates, the software allows/disables the generation of UEV events by means of this bit. 1: UEV is disabled. no update event is generated and the registers (ARR, PSC, CCRx) keep their values. If the UG bit is set or a hardware reset is issued from the mode controller, the counters and prescaler are reinitialized. 0: UEV is allowed. update (UEV) events are generated by any of the following events: -Counter overflow/underflow -Setting the UG position - Updates generated by the slave mode controller Registers with caches are loaded with their preloaded values.	0
0	CEN	RW	Enables the counter. 1: Enable the counter. 0: Disable the counter. *Note: The external clock, gated mode and encoder mode will not work until the CEN bit is set in software. Trigger mode can automatically set the CEN bit in hardware.*	0

9.4.2 Control Register 2 (TIM1_CTLR2)

Offset address: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Reserved	CCPC

Bit	Name	Access	Description	Reset value
[15:14]	Reserved	RO	Reserved	0
13	OIS3N	RW	Output idle state 3.	0

			1: When MOE=0, OC3N=1 after dead-time. 0: When MOE=0, OC3N=0 after dead-time. *Note: This bit cannot be modified after LOCK (TIMx_BDTR register) level 1, 2 or 3 has been set.*	
12	OIS3	RW	Output idle state 3. 1: When MOE=0, if OC3N is implemented, OC3=1 after dead-time. 0: When MOE=0, if OC3N is implemented, OC3=0. *Note: This bit cannot be modified after the LOCK (TIMx_BDTR register) level 1, 2 or 3 has been set.*	0
11	OIS2N	RW	Output idle state 2, see OIS3N.	0
10	OIS2	RW	Output idle state 2, see OIS3.	0
9	OIS1N	RW	Output idle state 1, see OIS3N.	0
8	OIS1	RW	Output idle state 1, see OIS3.	0
7	TI1S	RW	TI1 selection 1: TIMx_CH1, TIMx_CH2 and TIMx_CH3 pins connected to TI1 input (XOR combination). 0: TIMx_CH1 pin is connected directly to TI1 input.	0
[6:4]	MMS[2:0]	RW	Master mode selection: These 3 bits are used to select the synchronization information (TRGO) sent to the slave timer in master mode. Possible combinations are as follows: 000: Reset – The UG bit of the TIM1_EGR register is used as the trigger output (TRGO). If the reset is generated by the trigger input (the slave mode controller is in reset mode), there will be a delay between the signal on TRGO and the actual reset; 001: Enable – The counter enable signal CNT_EN is used as the trigger output (TRGO). Sometimes it is necessary to start multiple timers at the same time or control the slave timer to be enabled within a period of time. The counter enable signal is generated by the logical OR of the CEN control bit and the trigger input signal in gate mode. When the counter enable signal is controlled by the trigger input, there will be a delay on TRGO, unless the master/slave mode is selected (see the description of the MSM bit in the TIM1_SMCFGR register); 010: Update – The update event is selected as the trigger input (TRGO). For example, a master timer's clock can be used as a prescaler for a slave timer; 011: Compare pulse – When a capture occurs or a comparison is successful, when the CC1IF flag is set (even if it is already high), the trigger output sends a positive pulse (TRGO); 100: The compare-OC1REF signal is used as a trigger output (TRGO); 101: The compare-OC2REF signal is used as a trigger output (TRGO); 110: The compare–OC3REF signal is used as a trigger output (TRGO); 111: Reserved.	0
3	CCDS	RW	Capture compare DMA selection 1: Send a DMA request for CHxCVR when an update event occurs. 0: Generate a DMA request for CHxCVR when CHxCVR occurs.	0

2	CCUS	RW	Compare capture control update selection bits. 1: If CCPC is set, they can be updated by setting the COM bit or a rising edge on TRGI. 0: If the CCPC is set, they can only be updated by setting the COM bit. *Note: This bit only works for channels with complementary outputs.*	0
1	Reserved	RO	Reserved	0
0	CCPC	RW	Compare capture preload control bits. 1: The CCxE, CCxNE and OCxM bits are preloaded and when this bit is set they are only updated when the COM bit is set. 0: CCxE, CCxNE and OCxM bits are not preloaded. *Note: This bit only works for channels with complementary outputs.*	0

9.4.3 Slave Mode Control Register (TIM1_SMCFGR)

Offset address: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Reserved	SMS[2:0]		

Bit	Name	Access	Description	Reset value
15	ETP	RO	ETR trigger polarity selection, this bit selects whether to input ETR directly or to input the inverse of ETR. 1: Invert ETR, low or falling edge active; 0: ETR, active high or rising edge.	0
14	ECE	RW	External clock mode 2 enable selection. 1: Enables external clock mode 2. 0: Disable external clock mode 2. *Note 1: Slave mode can be used simultaneously with external clock mode 2: reset mode, gated mode and trigger mode; however, TRGI cannot be connected to ETRF in this case (TS bit cannot be '111').* *Note 2: When both external clock mode 1 and external clock mode 2 are enabled, the external clock input is ETRF.*	0
[13:12]	ETPS[1:0]	RW	The external trigger signal (ETRP) divides the frequency of this signal, which cannot exceed a maximum of 1/4 of the TIMxCLK frequency, and can be downconverted through this domain. 00: Prescaler off. 01: ETRP frequency divided by 2. 10: ETRP frequency divided by 4. 11: ETRP frequency divided by 8.	0
[11:8]	ETF[3:0]	RW	Externally triggered filtering, in fact, the digital filter is an event counter, which uses a certain sampling frequency to record up to N events and then produces a jump in the output. 0000: No fliter, sample as Fdts; 0001: Sampling frequency Fsampling=Fck_int, N=2; 0010: Sampling frequency Fsampling=Fck_int, N=4; 0011: Sampling frequency Fsampling=Fck_int, N=8; 0100: Sampling frequency Fsampling=Fdts/2, N=6; 0101: Sampling frequency Fsampling=Fdts/2, N=8; 0110: Sampling frequency Fsampling=Fdts/4, N=6;	0

			0111: Sampling frequency Fsampling=Fdts/4, N=8; 1000: Sampling frequency Fsampling=Fdts/8, N=6; 1001: Sampling frequency Fsampling=Fdts/8, N=8; 1010: Sampling frequency Fsampling=Fdts/16, N=5; 1011: Sampling frequency Fsampling=Fdts/16, N=6; 1100: Sampling frequency Fsampling=Fdts/16, N=8; 1101: Sampling frequency Fsampling=Fdts/32, N=5; 1110: Sampling frequency Fsampling=Fdts/32, N=6; 1111: Sampling frequency Fsampling=Fdts/32, N=8.	
7	MSM	RW	Master/slave mode selection. 1: The event on the trigger input (TRGI) is delayed to allow perfect synchronization between the current timer (via TRGO) and its slave timer. This is useful when the synchronization of several timers to a single external event is required. 0: Does not function.	0
[6:4]	TS[2:0]	RW	Trigger selection field, these 3 bits select the trigger input source used to synchronize the counter. 000: Internal trigger 0 (ITR0). 001: Internal trigger 1 (ITR1). 010: Internal trigger 2 (ITR2). 011: Internal trigger 3 (ITR3). 100: Edge detector of TI1 (TI1F_ED). 101: Filtered timer input 1 (TI1FP1). 110: Filtered timer input 2 (TI2FP2). 111: External trigger input (ETRF). The above only changes when SMS is 0. *Note: See Table 10-2 for details.*	0
3	Reserved	RO	Reserved	0
[2:0]	SMS[2:0]	RW	Input mode selection field. Selects the clock and trigger mode of the core counter. 000: Driven by the internal clock CK_INT. 001: Encoder mode 1, where the core counter increments or decrements the count at the edge of TI2FP2 depending on the level of TI1FP1. 010: Encoder mode 2, where the core counter increments or decrements the count at the edge of TI1FP1, depending on the level of TI2FP2. 011: Encoder mode 3, where the core counter increments and decrements the count on the edges of TI1FP1 and TI2FP2 depending on the input level of another signal; 100: reset mode, where the rising edge of the trigger input (TRGI) will initialize the counter and generate a signal to update the registers. 101: Gated mode, when the trigger input (TRGI) is high, the counter clock is turned on; at the trigger input becomes low, the counter is stopped, and the counter starts and stops are controlled. 110: Trigger mode, where the counter is started on the rising edge of the trigger input TRGI and only the start of the counter is controlled. 111: External clock mode 1, rising edge of the selected trigger input (TRGI) drives the counter.	0

9.4.4 DMA/Interrupt Enable Register (TIM1_DMAINTENR)

Offset address: 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	TDE	COMDE	Reserved	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	Reserved	CC3IE	CC2IE	CC1IE	UIE

Bit	Name	Access	Description	Reset value
15	Reserved	RO	Reserved	0
14	TDE	RW	Trigger DMA request enable bit. 1: Enable triggering of DMA requests; 0: Disable triggering of DMA requests.	0
13	COMDE	RW	COM DMA request enable bit. 1: Enable COM DMA requests; 0: Disable COM DMA requests.	0
12	Reserved	RO	Reserved	0b
11	CC3DE	RW	Compare/capture channel 3 DMA request enable bit. 1: Enable the DMA request of compare capture channel 3; 0: Disable the DMA request of compare capture channel 3.	0
10	CC2DE	RW	Compare/capture channel 2 DMA request enable bit. 1: Enable the DMA request of compare capture channel 2; 0: Disable the DMA request of compare capture channel 2.	0
9	CC1DE	RW	Compare/capture channel 1 DMA request enable bit. 1: Enable the DMA request of compare capture channel 1; 0: Disable the DMA request of compare capture channel 1.	0
8	UDE	RW	Update DMA request enable bit. 1: Enable the updated DMA request 0: Disable the updated DMA request.	0b
7	BIE	RW	Brake interrupt enable bit. 1: Enable the brake interrupt; 0: Disable the brake interrupt.	0
6	TIE	RW	Trigger interrupt enable bit 1: Enable the trigger interrupt; 0: Disable the trigger interrupt.	0
5	COMIE	RW	COM interrupt enable bit 1: Enable COM interrupt; 0: Disable COM interrupt.	0
4	Reserved	RO	Reserved	0
3	CC3IE	RW	Compare capture channel 3 interrupt enable bit 1: Enable compare capture channel 3 interrupt; 0: Disable compare capture channel 3 interrupt.	0
2	CC2IE	RW	Compare capture channel 2 interrupt enable bit 1: Enable compare capture channel 2 interrupt; 0: Disable compare capture channel 2 interrupt.	0
1	CC1IE	RW	Compare capture channel 1 interrupt enable bit 1: Enable compare capture channel 1 interrupt; 0: Disable compare capture channel 1 interrupt.	0
0	UIE	RW	Update interrupt enable bit 1: Enable updated interrupt; 0: Disable updated interrupt.	0

9.4.5 Interrupt Status Register (TIM1_INTFR) (x=1/2)

Offset address: 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reserved	CC3OF	CC2OF	CC1OF	Reserved	BIF	TIF	COMIF	Reserved	CC3IF	CC2IF	CC1IF	UIF

Bit	Name	Access	Description	Reset value
[15:12]	Reserved	RO	Reserved	0
11	CC3OF	RW0	Compare capture channel 3 over-capture flag bit.	0
10	CC2OF	RW0	Compare capture channel 2 over-capture flag bit.	0
9	CC1OF	RW0	Compare capture channel 1 over-capture flag, only used when the compare capture channel is configured in input capture mode. This flag is set by hardware and cleared by software by writing 0. 1: When the counter value is captured into the capture compare register, the status of CC1IF has been set; 0: No over-capture occurs.	0b
8	Reserved	RO	Reserved	0
7	BIF	RW0	The brake interrupt flag bit, once the brake input is valid, by hardware for this position bit, can be cleared by software. 1: A set valid level is detected on the brake pin input. 0: No braking event is generated.	0
6	TIF	RW0	Trigger interrupt flag bit, when a trigger event occurs by hardware to this location bit, by software to clear. Trigger events include the detection of a valid edge at the TRGI input from a mode other than gated, or any edge in gated mode. 1: Trigger event generation. 0: No trigger event is generated.	0
5	COMIF	RW0	COM interrupt flag bit, this bit is set by hardware and cleared by software once a COM event is generated. com events including CCxE, CCxNE, OCxM are updated. 1: COM event generation. 0: No COM event is generated.	0
4	Reserved	RO	Reserved	0
3	CC3IF	RW0	Compare capture channel 3 interrupt flag bit.	0
2	CC2IF	RW0	Compare capture channel 2 interrupt flag bit.	0
1	CC1IF	RW0	Compare capture channel 1 interrupt flag bit. If the compare capture channel is configured in output mode. This bit is set by hardware when the counter value matches the comparison value, except in centrosymmetric mode. This bit is cleared by software. 1: The value of the core counter matches the value of compare capture register 1; 0: No match occurs. If compare capture channel 1 is configured as input mode. This bit is set by hardware when a capture event occurs, and it is cleared by software or by reading the compare capture register. 1: The counter value has been captured compare capture register 1. 0: No input capture is generated.	0
0	UIF	RW0	Update interrupt flag bit, this bit is set by hardware when an update event is generated and cleared by software. 1: Update interrupt generation. 0: No update event is generated. The following scenarios generate update events. If UDIS = 0, when the repeat counter value overflows or	0

			underflows. If URS = 0, UDIS = 0, when the UG bit is set, or when the counter core counter is reinitialized by software. If URS = 0, UDIS = 0, when the counter CNT is reinitialized by a trigger event.	

9.4.6 Event Generation Register (TIM1_SWEVGR)

Offset address: 0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								BG	TG	COMG	Reserved	CC3G	CC2G	CC1G	UG

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
7	BG	WO	The brake event generation bit, which is set and cleared by software, is used to generate a brake event. 1: Generate a brake event. At this point, MOE=0, BIF=1, if the corresponding interrupt and DMA are enabled, the corresponding interrupt and DMA are generated. 0: No action.	0
6	TG	WO	The trigger event generation bit, which is set by software and cleared by hardware, is used to generate a trigger event. 1: Generate a trigger event, TIF is set, and the corresponding interrupts and DMAs are generated if enabled. 0: No action.	0
5	COMG	WO	Compare capture control update generation bit. Generates a compare capture control update event. This bit is set by software and automatically cleared by hardware. 1: When CCPC = 1, allow updating of CCxE, CCxNE, OCxM bits. 0: No action. *Note: This bit is only valid for channels with complementary outputs (channels 1, 2, 3).*	0
4	Reserved	RO	Reserved	0
3	CC3G	WO	Compare capture event generation bit 3. generates compare capture event 3.	0
2	CC2G	WO	Compare capture event generation bit 2. generates compare capture event 2.	0
1	CC1G	WO	Compare capture event generation bit 1. generates compare capture event 1. This bit is set by software and cleared by hardware. It is used to generate a compare capture event. 1: Generate a compare capture event on compare capture channel 1. If compare capture channel 1 is configured as output. Set the CC1IF bit. Generate the corresponding interrupts and DMAs if they are enabled. If compare capture channel 1 is configured as input. The current core counter value is captured to compare capture register 1; set the CC1IF bit to generate the corresponding interrupts and DMAs if they are enabled;	0

			if CC1IF is already set, set the CC1OF bit. 0: No action.	
0	UG	WO	Update event generation bit to generate an update event. This bit is set by software and is automatically cleared by hardware. 1: Initialize the counter and generate an update event. 0: No action. *Note: The prescaler counter is also cleared to zero, but the prescaler factor remains unchanged. The core counter is cleared if in centrosymmetric mode or incremental counting mode; if in decremental counting mode, the core counter takes the value of the reload value register.*	0

9.4.7 Compare/Capture Control Register 1 (TIM1_CHCTLR1)

Offset address: 0x18

The channel can be used in input (capture mode) or output (compare mode), and the direction of the channel is defined by the corresponding CCxS bit. The other bits of this register have different roles in input and output modes. OCxx describes the function of the channel in output mode and ICxx describes the function of the channel in input mode.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			

Compare mode (pin direction is output).

Bit	Name	Access	Description	Reset value
15	OC2CE	RW	Compare capture channel 2 clear enable bit. 1: Clear OC2REF bit zero once ETRF input is detected high; 0: OC2REF is not affected by ETRF input.	0
[14:12]	OC2M[2:0]	RW	Compare Capture Channel 2 mode setting field. The 3 bits define the action of the output reference signal OC2REF, which determines the values of OC2, OC2N. OC2REF is active high, while the active levels of OC2 and OC2N depend on the CC2P, CC2NP bits. 000: Freeze. The comparison value between the value of the capture register and the core counter has no effect on OC2REF; 001: Forced to valid level. When the core counter has the same value as compare capture register 2, force OC2REF to be high; 010: Forced to invalid level. When the value of the core counter is the same as compare capture register 2, force OC2REF low; 011: Flip. When the value of the core counter and compare capture register 2 are the same, flip the level of OC2REF; 100: Forced to invalid level. Force OC2REF low; 101: Forced to valid level. Force OC2REF high; 110: PWM mode 1: When counting up, once the core counter is less than the value of the comparison capture register, channel 2 is an effective level, otherwise it is an invalid level; When counting down, once the core	0

			counter is greater than the value of the comparison capture register, channel 2 is at an invalid level (OC2REF=0), otherwise it is at an effective level (Oc2Ref = 1); 111: PWM mode 2: When counting up, once the core counter is less than the value of the comparison capture register, channel 2 is at an invalid level, otherwise it is at an effective level; When counting down, once the core counter is greater than the value of the comparison capture register, Channel 2 is active (OC2REF=1), otherwise it is inactive (OC2REF=0). *Note: Once the LOCK level is set to 3 and CC2S=00b, this bit cannot be modified. In PWM mode 1 or PWM mode 2, the OC2REF level changes only when the comparison result changes or when switching from freeze mode to PWM mode in output compare mode.*	
11	OC2PE	RW	Compare capture register 2 preload enable bit. 1: Enable the preload function of compare capture register 2. Read and write operations only operate on the preload register. The preload value of compare capture register 2 is loaded into the current shadow register when the update event arrives; 0: Disable the preload function of compare capture register 2, you can write to compare capture register 2 at any time, and the newly written value will take effect immediately. *Note: Once the LOCK level is set to 3 and CC2S=00, this bit cannot be modified; only in single pulse mode (OPM=1), the PWM mode can be used without confirming the preload register, otherwise its action is uncertain.*	0
10	OC2FE	RW	Compare capture channel 2 fast enable bit, which is used to speed up the response of the compare capture channel output to trigger input events. 1: The valid edge input to the flip-flop acts as if a compare match occurred. Therefore, OC is set to the comparison level regardless of the comparison result. The delay between the valid edge of the sampling flip-flop and the output of compare capture channel 2 is shortened to 3 clock cycles; 0: Based on the value of the counter and compare capture register 1, compare capture channel 2 operates normally, even if the flip-flop is open. The minimum delay to activate the compare capture channel 2 output is 5 clock cycles when the input of the flipflop has a valid edge. OC2FE only works when the channel is configured to PWM1 or PWM2 mode.	0
[9:8]	CC2S[1:0]	RW	Compare capture channel 2 input selection fields. 00: Compare capture channel 2 is configured as an output. 01: Compare capture channel 2 is configured as an input and IC2 is mapped on TI2. 10: Compare capture channel 2 is configured as an input and IC2 is mapped on TI1. 11: Compare Capture Channel 2 is configured as an input and IC2 is mapped on TRC. This mode works only when the internal trigger input is selected (by the	0

			TS bit). *Note: Compare Capture Channel 2 is writable only when the channel is off (when CC2E is zero).*	
7	OC1CE	RW	Compare capture channel 1 clear enable bit.	0
[6:4]	OC1M[2:0]	RW	Compare capture channel 1 mode setting field.	0
3	OC1PE	RW	Compare capture register 1 preload enable bit.	0
2	OC1FE	RW	Compare capture channel 1 fast enable bit.	0
[1:0]	CC1S[1:0]	RW	Compare capture channel 1 input selection fields.	0

Capture mode (pin direction is input).

Bit	Name	Access	Description	Reset value
[15:12]	IC2F[3:0]	RW	The input capture filter 2 configuration field, these bits set the sampling frequency of the TI1 input and the digital filter length. The digital filter consists of an event counter, which records N events and then generates a jump in the output. 0000: No filter, sampled at fDTS. 1000: Sampling frequency Fsampling = Fdts/8, N = 6. 0001: Sampling frequency Fsampling=Fck_int, N=2. 1001: Sampling frequency Fsampling = Fdts/8, N = 8. 0010: Sampling frequency Fsampling=Fck_int, N=4. 1010: Sampling frequency Fsampling = Fdts/16, N = 5. 0011: Sampling frequency Fsampling=f=Fck_int, N=8. 1011: Sampling frequency Fsampling = Fdts/16, N = 6. 0100: Sampling frequency Fsampling = Fdts/2, N = 6. 1100: Sampling frequency Fsampling = Fdts/16, N = 8. 0101: Sampling frequency Fsampling = Fdts/2, N = 8. 1101: Sampling frequency Fsampling = Fdts/32, N = 5. 0110: Sampling frequency Fsampling = Fdts/4, N = 6. 1110: Sampling frequency Fsampling = Fdts/32, N = 6. 0111: Sampling frequency Fsampling = Fdts/4, N = 8. 1111: Sampling frequency Fsampling=Fdts/32, N=8.	0
[11:10]	IC2PSC[1:0]	RW	Compare capture channel 2 prescaler configuration field, these 2 bits define the prescaler coefficient for compare capture channel 2. Once CC1E = 0, the prescaler is reset. 00: Without prescaler, one capture is triggered for each edge detected on the capture input. 01: Capture triggered every 2 events. 10: Capture triggered every 4 events. 11: Capture is triggered every 8 events.	0
[9:8]	CC2S[1:0]	RW	Compare the capture channel 2 input selection field, these 2 bits define the direction of the channel (input/output), and the selection of the input pin. 00: Compare capture channel 1 channel is configured as an output. 01: Compare capture channel 1 channel is configured as an input and IC1 is mapped on TI1. 10: Compare capture channel 1 channel is configured as an input and IC1 is mapped on TI2. 11: Compare capture channel 1 channel is configured as an input and IC1 is mapped on TRC. This mode works only when the internal trigger input is selected (by the TS bit). *Note: CC1S is writable only when the channel is off (CC1E is 0).*	0

[7:4]	IC1F[3:0]	RW	Input capture filter 1 configuration field.	0
[3:2]	IC1PSC[1:0]	RW	Compare capture channel 1 prescaler configuration field.	0
[1:0]	CC1S[1:0]	RW	Compare capture channel 1 input selection field.	0

9.4.8 Compare/Capture Control Register 2 (TIM1_CHCTLR2)

Offset address: 0x1C

Channels can be used as input (capture mode) or output (compare mode), and the direction of the channel is defined by the corresponding CCxS bit. The functions of other bits in this register are different in input and output modes. OCxx describes the function of the channel in output mode, and ICxx describes the function of the channel in input mode.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
								IC3F[3:0]				IC3PSC[1:0]			

Compare mode (pin direction is output).

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
7	OC3CE	RW	Compare capture channel 3 clear enable bit.	0
[6:4]	OC3M[2:0]	RW	Compare capture channel 3 mode setting fields.	0
3	OC3PE	RW	Compare capture register 3 preload enable bit.	0
2	OC3FE	RW	Compare capture channel 3 fast enable bit.	0
[1:0]	CC3S[1:0]	RW	Compare capture channel 3 input selection field.	0

Capture mode (pin direction is input).

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
[7:4]	IC3F[3:0]	RW	Input capture filter 3 configuration field.	0
[3:2]	IC3PSC[1:0]	RW	Compare capture channel 3 prescaler configuration field.	0
[1:0]	CC3S[1:0]	RW	Compare capture channel 3 input selection field.	0

9.4.9 Compare/Capture Enable Register (TIM1_CCER)

Offset address: 0x20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E

Bit	Name	Access	Description	Reset value
[15:12]	Reserved	RO	Reserved	0
11	CC3NP	RW	Compare capture channel 3 complementary output polarity setting bit.	0
10	CC3NE	RW	Compare capture channel 3 complementary output enable bit.	0
9	CC3P	RW	Compare capture channel 3 output polarity setting bit.	0
8	CC3E	RW	Compare capture channel 3 output enable bit.	0
7	CC2NP	RW	Compare capture channel 2 complementary output polarity setting bit.	0
6	CC2NE	RW	Compare capture channel 2 complementary output	0

			enable bit.	
5	CC2P	RW	Compare capture channel 2 output polarity setting bit. *Note: When using the complementary output function, use bit6 and bit7.*	0
4	CC2E	RW	Compare capture channel 2 output enable bit.	0
3	CC1NP	RW	Compare capture channel 1 complementary output polarity setting bit.	0
2	CC1NE	RW	Compare capture channel 1 complementary output enable bit.	0
1	CC1P	RW	Compare capture channel 1 output polarity setting bit. CC1 channel is configured as output: 1: OC1 is active at low level; 0: OC1 is active at high level. CC1 channel configured as input: This bit selects whether IC1 or the inverted signal of IC1 is used as the trigger or capture signal. 1: Invert: Capture occurs on the falling edge of IC1; when used as an external trigger, IC1 is inverted. 0: No inversion: capture occurs on the rising edge of IC1; when used as an external trigger, IC1 does not invert. *Note: Once the LOCK level (LOCK bit in the TIM1_BDTR register) is set to 3 or 2, this bit cannot be modified.* *Note: When using the complementary output function, bit3 and bit2 are used.*	0
0	CC1E	RW	Compare capture channel 1 output enable bit. CC1 channel configured as output: 1: On. OC1 signals are output to the corresponding output pins, and its output level depends on the values of the MOE, OSSI, OSSR, OIS1, OIS1N, and CC1NE bits. 0: Off. OC1 disables the output, so the output level of OC1 depends on the values of the MOE, OSSI, OSSR, OIS1, OIS1N and CC1NE bits. CC1 channel is configured as input: this bit determines whether the counter value can be captured into the TIM1_CCR1 register. 1: Capture enabled; 0: Capture disabled.	0

9.4.10 Counter for Advanced-control Timer (TIM1_CNT)

Offset address: 0x24

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	CNT[15:0]	RW	The real-time value of the timer's counter.	0

9.4.11 Counting Clock Prescaler (TIM1_PSC)

Offset address: 0x28

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

PSC[15:0]

Bit	Name	Access	Description	Reset value
[15:0]	PSC[15:0]	RW	The division coefficient of the timer's prescaler; the clock frequency of the counter is equal to the input frequency of the divider/(PSC+1).	0

9.4.12 Auto-reload Value Register (TIM1_ATRLR)

Offset address: 0x2C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ATRLR[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	ATRLR[15:0]	RW	The value of this field will be loaded into the counter. When ATRLR acts and is updated, see Chapter 9.2.3; when ATRLR is empty, the counter stops.	0

9.4.13 Repeat Count Value Register (TIM1_RPTCR)

Offset address: 0x30

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								RPTCR[7:0]							

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
[7:0]	RPTCR[7:0]	RW	The value of the repeat counter.	0

9.4.14 Compare/Capture Register 1 (TIM1_CH1CVR)

Offset address: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL 1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH1CVR[15:0]															

Bit	Name	Access	Description	Reset value
[31:17]	Reserved	RO	Reserved	0
16	LEVEL1	RO	The level indication bit corresponding to the captured value	0
[15:0]	CH1CVR[15:0]	RW	The value of compare capture register channel 1.	0

9.4.15 Compare/Capture Register 2 (TIM1_CH2CVR)

Offset address: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL

															2
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH2CVR[15:0]															

Bit	Name	Access	Description	Reset value
[31:17]	Reserved	RO	Reserved	0
16	LEVEL2	RO	The level indication bit corresponding to the captured value	0
[15:0]	CH2CVR[15:0]	RW	The value of compare capture register channel 2.	0

9.4.16 Compare/Capture Register 3 (TIM1_CH3CVR)

Offset address: 0x3C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL 3
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3CVR[15:0]															

Bit	Name	Access	Description	Reset value
[31:17]	Reserved	RO	Reserved	0
16	LEVEL3	RO	The level indication bit corresponding to the captured value	0
[15:0]	CH3CVR[15:0]	RW	The value of compare capture register channel 3.	0

9.4.17 Brake and Dead-time Register (TIM1_BDTR)

Offset address: 0x44

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							

Bit	Name	Access	Description	Reset value
15	MOE	RW	Main output enable bit. Once the brake signal is active, it will be cleared asynchronously. 1: Enable OCx and OCxN to be set as outputs. 0: Disable the output of OCx and OCxN or force to idle state.	0
14	AOE	RW	Auto output enable. 1: MOE can be set by software or set in the next update event. 0: MOE can only be set by software.	0
13	BKP	RW	The brake input polarity setting bit. 1: Brake input active high. 0: Brake input active low. *Note: When LOCK level 1 is set, this bit cannot be modified. A write to this bit requires an HB clock before it can take effect.*	0
12	BKE	RW	Brake function enable bit. 1: Brake input enabled. 0: Brake input disabled.	0

			Note: When LOCK level 1 is set, this bit cannot be modified. A write to this bit requires an HB clock before it can take effect.	
11	OSSR	RW	1: When the timer is not operating, once CCxE=1 or CCxNE=1, first turn on OC/OCN and outputinvalid level, then set OCx, OCxN enable output signal=1. 0: When the timer is not operating, OC/OCN output is disabled. *Note: When LOCK level 1 is set, this bit cannot be modified.*	0
10	OSSI	RW	1: When the timer is not operating, once CCxE = 1 or CCxNE = 1, OC/OCN first outputs its idle level, then OCx, OCxN enable output signal = 1. 0: When the timer is not operating, OC/OCN output is disabled. *Note: When LOCK level 1 is set, this bit cannot be modified.*	0
[9:8]	LOCK[1:0]	RW	Lock the function setting field. 00: Disable the locking function. 01: Lock level 1, no DTG, BKE, BKP, AOE, OISx and OISxN bits can be written. 10: Lock level 2, where the bits in lock level 1 cannot be written, nor the CC polarity bits, nor the OSSR and OSSI bits. 11: Lock level 3, cannot write to the bits in lock level 2, and cannot write to the CC control bits. *Note: After system reset, the LOCK bit can only be written once and cannot be modified again until reset.*	0
[7:0]	DTG[7:0]	RW	Dead-band setting bits, these bits define the dead-band duration between complementary outputs. Assume DT represents its duration: DTG[7:5]=0xx=>DT=DTG[7:0]*Tdtg, Tdtg =TDTS; DTG[7:5]=10x=>DT=(64+DTG[5:0])*Tdtg, Tdtg= 2*TDTS; DTG[7:5]=110=>DT=(32+DTG[4:0])*Tdtg, Tdtg =8 ×TDTS; DTG[7:5]=111=>DT=(32+DTG[4:0])*Tdtg, Tdtg =16 *TDTS. *Note: Once the LOCK level (LOCK[1:0] bits in the TIM1_BDTR register) is set to 01b, 10b, or 11b, these bits cannot be modified.*	0

9.4.18 DMA Control Register (TIM1_DMACFGR)

Offset address: 0x48

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			DBL[4:0]					Reserved			DBA[4:0]				

Bit	Name	Access	Description	Reset value
[15:13]	Reserved	RO	Reserved	0
[12:8]	DBL[4:0]	RW	The length of DMA continuous transfer. The actual value is the value of this field + 1.	0
[7:5]	Reserved	RO	Reserved	0
[4:0]	DBA[4:0]	RW	These bits define the offset of the DMA from the address of control register 1 in continuous mode.	0

9.4.19 DMA Address Register for Continuous Mode (TIM1_DMAADR)

Offset address: 0x4C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	DMAB[15:0]	RW	The address of the DMA in continuous mode.	0

Chapter 10 General-Purpose Timer (GPTM)

The general-purpose timer module contains a 16-bit auto-reloadable timer (TIM2), for measuring pulse width or generating pulses of a specific frequency, PWM waves, etc. It can be used in automation control, power supply, etc.

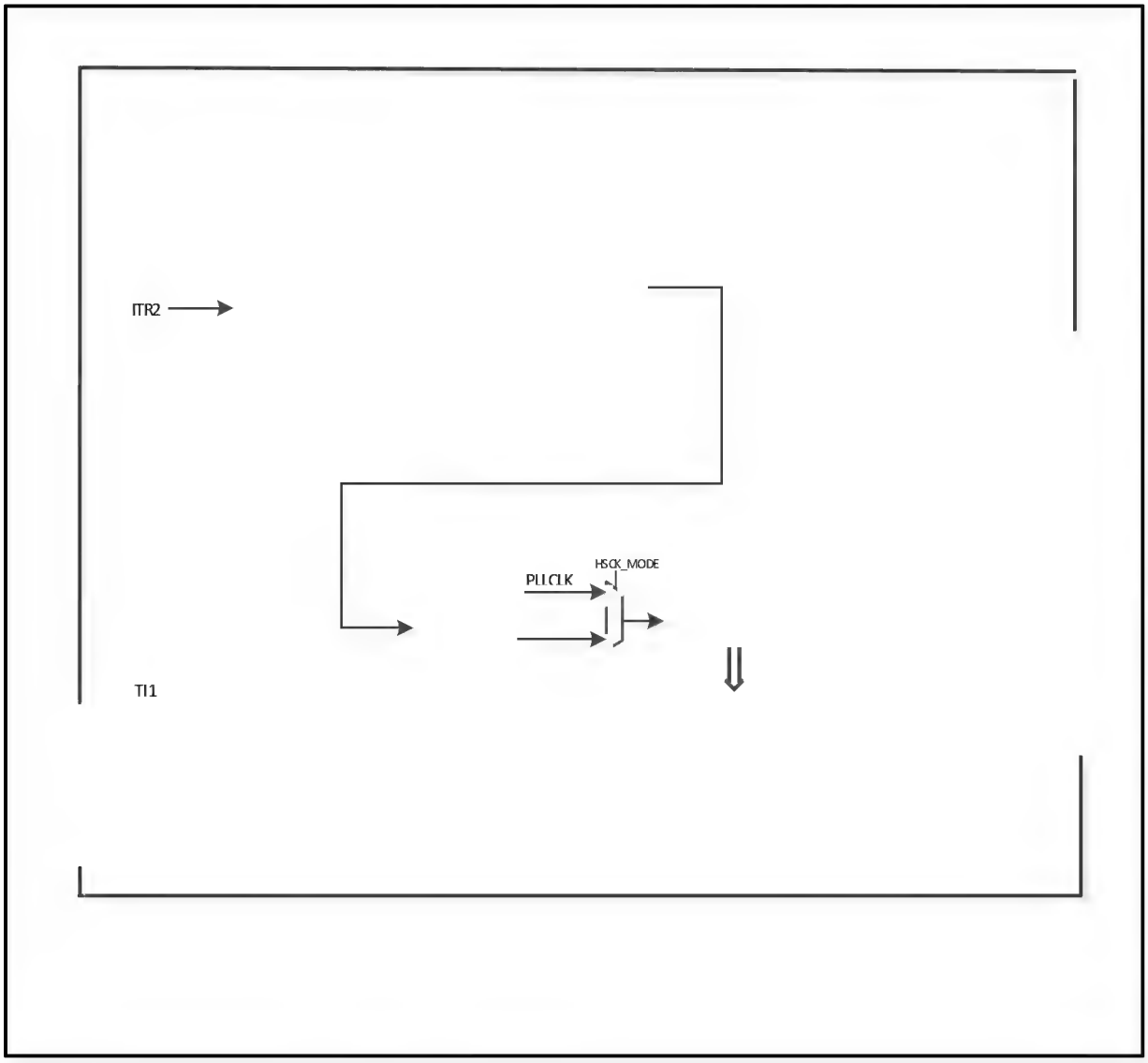
10.1 Main Features

The main features of the general-purpose timer include:

- 16-bit auto-reload counter, supports incremental counting mode
- 16-bit prescaler with dynamically adjustable crossover factor from 1 to 65536
- Support 2 independent comparison capture channels
- Each compare capture channel supports multiple operating modes, such as: input capture, output compare, PWM generation, and single pulse output
- Support external signal control timer
- Support cascading and synchronization between timers
- Support simple dead-time control

10.2 Principle and Structure

Figure 10-1 Block diagram of the structure of the general-purpose timer



10.2.1 Overview

As shown in Figure 10-1, the structure of the general timer can be roughly divided into three parts, namely the input clock part, the core counter part and the compare capture channel part.

The clock of the general timer can come from the HB bus clock (CK_INT), from other timers with clock output function (ITRx), or from the input end of the compare capture channel (TIM2_CHx). These input clock signals become the CK_PSC clock after various set filtering and frequency division operations, and are output to the core counter part.

The core of the general-purpose timer is a 16-bit counter (CNT). After CK_PSC is divided by the prescaler (PSC), it becomes CK_CNT and is finally lost to CNT. CNT supports up-counting mode and has an automatic reload value register (ATRLR) that reloads and initializes CNT after each counting cycle. value.

The general-purpose timer has two sets of compare capture channels. Each set of compare capture channels can input pulses from a dedicated pin or output a waveform to the pin. That is, the compare capture channel supports

input and output modes. The input of each channel of the compare capture register supports filtering, frequency division, edge detection and other operations, supports mutual triggering between channels, and can also provide a clock for the core counter CNT. Each compare capture channel has a set of compare capture registers (CHxCVR) that support comparison with the main counter (CNT) and output pulses.

10.2.2 Differences between General-Purpose Timer and Advanced-Control Timer

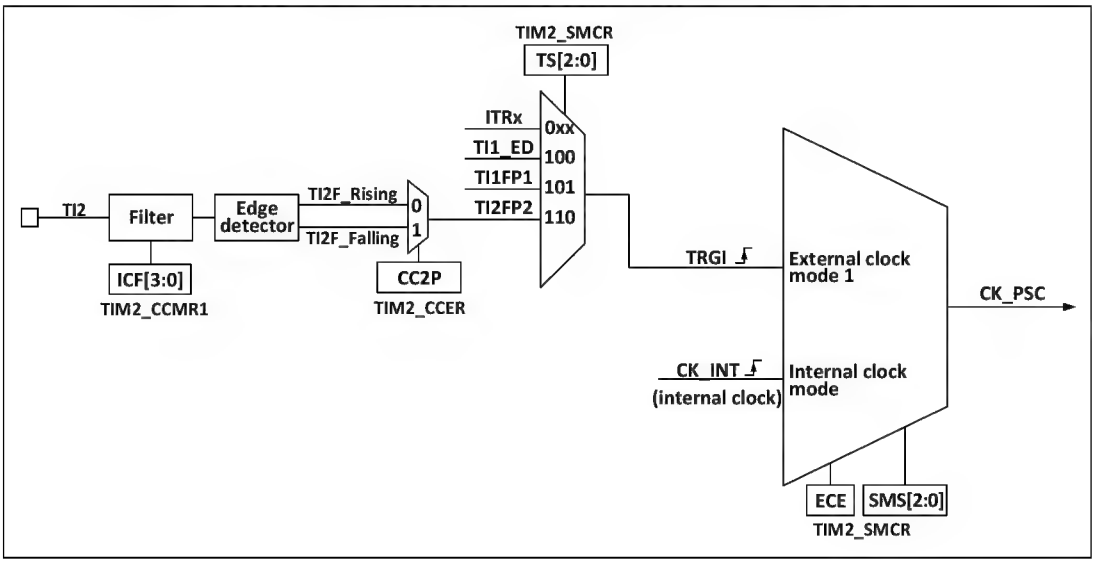
Compared to advanced-control timers, general-purpose timers lack the following features.

1) General-purpose timers lack a repeat count register for counting the count cycles of the core counter.
2) General-purpose timers do not have a brake signal mechanism.
3) General purpose timers do not have encoder mode.
4) General purpose timers do not support the use of DMA in multiple modes.
5) General timer 16-bit auto-reload counter does not support down-counting mode and up-down counting mode.
6) General-purpose timers do not support the external clock pin (ETR) input clock.

10.2.3 Clock Input

This section discusses the origin of CK_PSC. Here is the clock source part of the overall structure block diagram of the general-purpose timer.

Figure 10-2 General-Purpose timer CK_PSC source block diagram



The optional input clocks can be divided into 4 categories:

1) Internal HB clock input route: CK_INT.
2) Route from the compare capture channel pin (TIMx_CHx): TIMx_CHx → TIx → TIxFPx;
3) Input from other internal timers: ITRx.

The actual operation can be divided into 3 categories by determining the choice of input pulse for the SMS of the CK_PSC source.

1) Selection of the internal clock source (CK_INT).
2) External clock source mode 1.

All 2 clock source sources mentioned above can be selected by these 2 operations.

10.2.3.1 Internal Clock Source (CK_INT)

If the general-purpose timer is started when the SMS field is held at 000b, then it is the internal clock source (CK_INT) that is selected as the clock. At this point CK_INT is CK_PSC.

10.2.3.2 External Clock Source Mode 1

When the SMS domain is set to 111b, external clock source mode 1 is enabled. When external clock source 1 is enabled, TRGI is selected as the source for CK_PSC. it is worth noting that the user also needs to select the source for TRGI by configuring the TS domain. The TS domain can select the following types of pulses as clock sources.

1) Internal trigger (ITRx, x is 0,1,2,3).
2) Compare/capture the signal of channel 1 after passing through the edge detector (TI1F_ED);
3) Compare/capture channel signal TI1FP1, TI2FP2

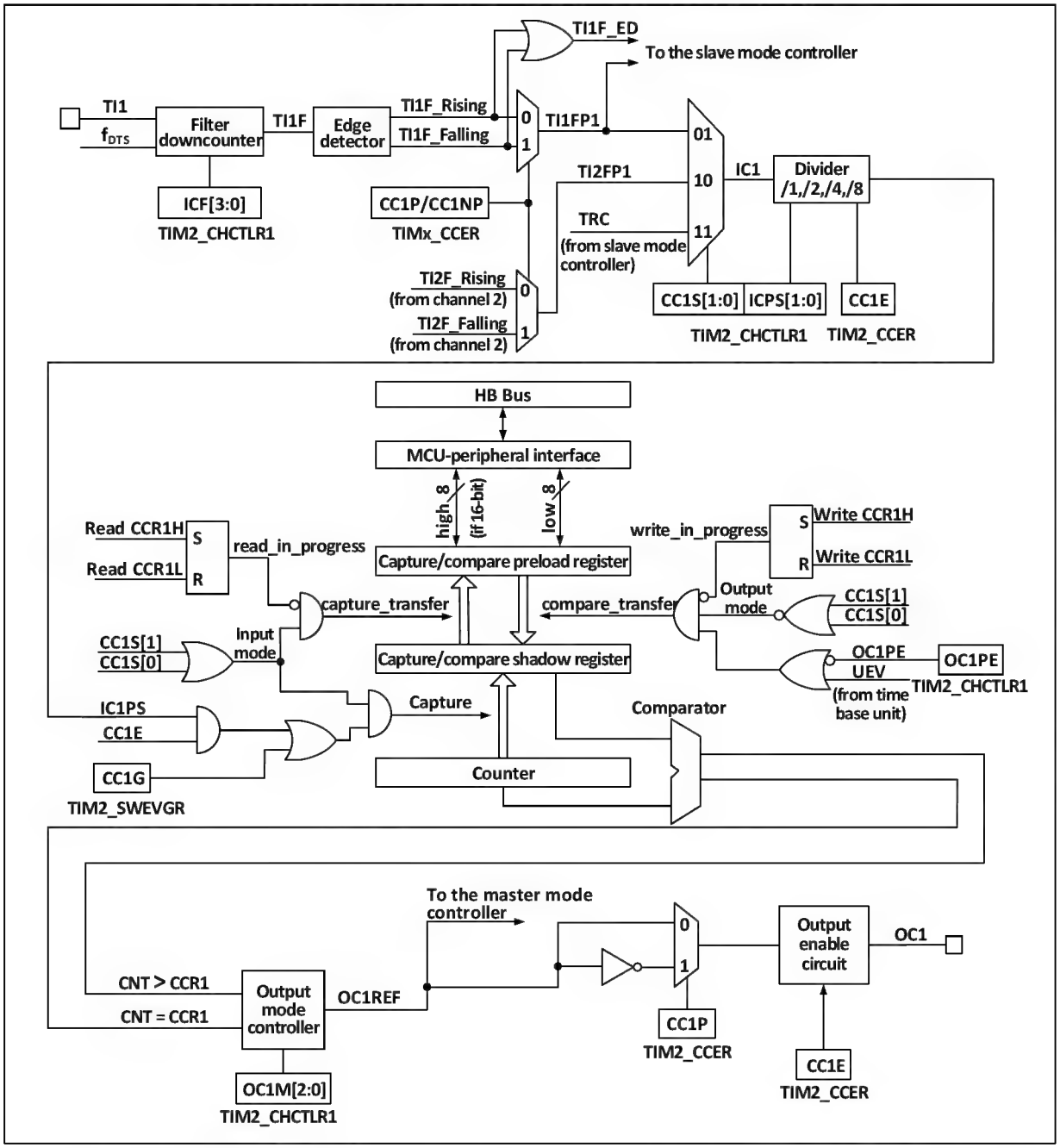
10.2.3 Counters and Peripherals

CK_PSC is input to the prescaler (PSC) for dividing the frequency, the PSC is 16-bit and the actual dividing factor is equal to the value of R16_TIM2_PSC + 1. CK_PSC becomes CK_INT after passing through the PSC. Changing the value of R16_TIM1_PSC will not take effect in real time, but will be updated to the PSC after an update event which includes the UG bit clear and reset. Update events include UG bit clear and reset.

10.2.4 Compare/Capture Channels

The compare capture channel is the core of the timer to achieve complex functions. Its core is the comparison capture register, supplemented by the digital filtering of the peripheral input part, frequency division and inter-channel multiplexing, and the comparator and output control of the output part. The structural block diagram of the compare capture channel is shown in Figure 10-3.

Figure 10-3 Block diagram of the structure of the compare capture channel



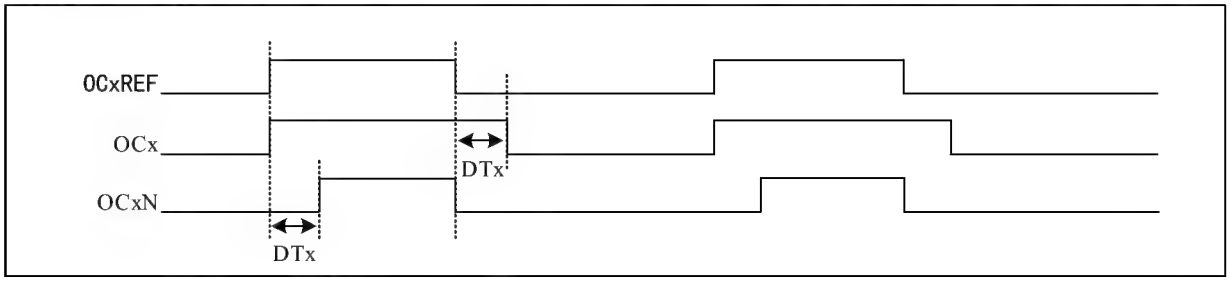
The signal is input from the channel x pin and optionally made as TIx (the source of TI1 can be more than CH1, see block diagram 10-1 of the timer), TI1 is passed through the filter (ICF[3:0]) to generate TI1F, and then divided into TI1F_Rising and TI1F_Falling through the edge detector, these two signals are selected (CC1P) to generate TI1FP1, TI1FP1 and TI2FP1 from channel 2 are sent together to CC1S to select to become IC1, which is sent to the compare capture register after ICPS dividing.

The compare capture register consists of a preload register and a shadow register, and the read/write process operates only on the preload register. In capture mode, the capture occurs on the shadow register and is then copied to the preload register; in compare mode, the contents of the preload register are copied to the shadow register, and then the contents of the shadow register are compared to the core counter (CNT).

10.2.5 Complementary Outputs and Dead-time

2 channels generally have two output pins, which can output two complementary signals with dead zones (OCx and

OCxN). The polarity is independently controlled through the COxP and COxNP bits, and the dead zone time can be set by setting the DTx bit.



10.3 Function and Implementation

The realization of the complex functions of the general timer is achieved by operating the timer's comparE capture channel, clock input circuit, counter and peripheral components. The clock input to the timer can come from multiple clock sources including the input to the compare capture channel. The operation of compare capture register channel and clock source selection directly determines its function. The compare capture channel is bidirectional and can work in input and output modes.

10.3.1 Input Capture Mode

Input capture mode is one of the basic functions of the timer. The principle of the input capture mode is that when a certain edge on the ICxPS signal is detected, a capture event is generated, and the current value of the counter will be latched into the comparison capture register (R16_TIM2_CHCTLRx). When a capture event occurs, CCxIF (in R16_TIM2_INTFR) is set, and if interrupts are enabled, corresponding interrupts will be generated. If CCxIF is already set when a capture event occurs, the CCxOF bit will be set. CCxIF can be cleared by software or by hardware by reading the compare capture register. CCxOF is cleared by software.

Take an example of channel 1 to illustrate the steps of using input capture mode, as follows:

1) Configure the CCxS domain and select the source of the ICx signal. For example, if it is set to 10b and TI1FP1 is selected as the source of IC1, the default settings cannot be used. The default setting of the CCxS domain is to use the comparison capture module as the output channel;
2) Configure the ICxF domain and set the digital filter for the TI signal. The digital filter will sample a certain frequency at a certain frequency, and then output a transition. This sampling frequency and number of times are determined by ICxF;
3) Configure the CCxP bit to set the polarity of TIxFPx. For example, keep the CC1P bit low and select rising edge transition;
4) Configure the ICxPS field to set the crossover coefficient between which the ICx signal becomes ICxPS. For example, keep ICxPS as 00b without crossover;
5) Configuring the CCxE bit allows capturing the value of the core counter (CNT) into the compare capture register. Set CC1E bit;
6) Configure the CCxIE and CCxDE bits as needed to decide whether to enable interrupts.

At this point, the compare capture channel configuration has been completed.

When TI1 inputs a captured pulse, the value of the core counter (CNT) will be recorded in the compare capture register, and CC1IF will be set. When CC1IF has been set before, the CCIOF bit will also be set. If CC1IE is bit, then an interrupt will be generated. An input capture event can be generated by software by writing to the event generation register (R16_TIM2_SWEVGR).

10.3.2 Compare Output Mode

Compare output mode is one of the basic functions of a timer. The principle of the compare output mode is to output a specific change or waveform when the value of the core counter (CNT) is consistent with the value of the compare capture register. The OCxM field (in R16_TIM2_CHCTLRx) and the CCxP bit (in R16_TIM2_CCER) determine whether the output is a definite high and low level or a level flip. When a comparison consistency event occurs, the CCxIF bit will also be set. If the CCxIE bit is set in advance, an interrupt will be generated.

The steps to configure compare output mode are as follows:

1) Configure the clock source and auto-reload value of the core counter (CNT);
2) Set the count value to be compared into the compare capture register (R16_TIM2_CHxCVR);
3) If an interrupt needs to be generated, set the CCxIE bit;
4) Keep OCxPE at 0 to disable the preload register of the compare capture register;
5) Set the output mode, set the OCxM field and CCxP bit;
6) Enable the output and set the CCxE bit;
7) Set the CEN bit to start the timer.

10.3.3 Forced Output Mode

The output mode of the timer's compare capture channel can be forced to output a determined level by software without relying on the comparison of the shadow register of the compare capture register and the core counter.

This is done by setting OCxM to 100b, which forces OCxREF to be set low, or by setting OCxM to 101b, which forces OCxREF to be set high.

It should be noted that when OCxM is forced to 100b or 101b, the comparison process between the internal main counter and the compare capture register is still in progress, the corresponding flag bit is still set, and interrupts are still generated.

10.3.4 PWM Input Mode

PWM input mode is used to measure the duty cycle and frequency of PWM. It is a special case of input capture mode. The operation is the same as input capture mode except for the following differences: PWM occupies two compare capture channels, and the input polarity of the two channels is set to opposite, one of the signals is set as the trigger input, and SMS is set in reset mode.

For example, to measure the period and frequency of the PWM wave input from TI1, you need to perform the following operations:

1) Set TI1 (TI1FP1) to be the input of IC1 signal. Set CC1S to 01b.
2) Set TI1FP1 to rising edge active. Holding CC1P at 0.
3) Set TI1 (TI1FP2) as the input of IC2 signal. Set CC2S to 10b.
4) Select TI1FP2 to set to falling edge active. Set CC2P to 1.
5) Select TI1FP1 as the source of the clock source. set TS to 101b.
6) Set the SMS to reset mode, i.e. 100b.
7) Enables input capture. CC1E and CC2E are set.

Note: Since only TI1FP1 and TI2FP2 are connected to the slave mode controller, only TIM2_CH1/TIM2_CH2 can be used in the PWM input mode.

10.3.5 PWM Output Mode

PWM output mode is one of the basic functions of the timer. The most common method of PWM output mode is to use the reload value to determine the PWM frequency and the capture compare register to determine the duty cycle.

Set the OCxM field to 110b or 111b to use PWM mode 1 or mode 2, set the OCxPE bit to enable the preload register, and finally set the ARPE bit to enable automatic reloading of the preload register. The value of the preload register can be sent to the shadow register when an update event occurs, so before the core counter starts counting, the UG bit needs to be set to initialize all registers.

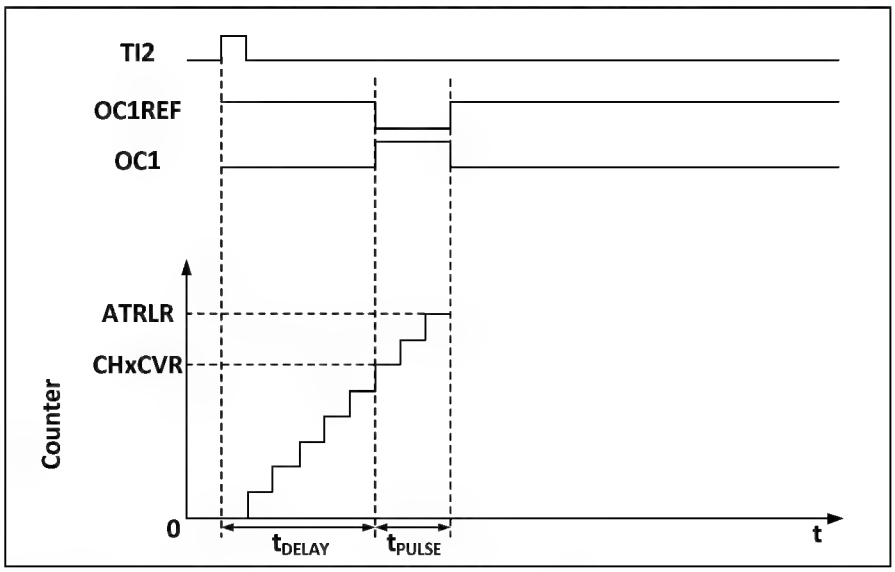
- Edge aligned

When using edge alignment, the core counter counts up. In the case of PWM mode 1, when the value of the core counter is greater than the compare capture register, OCxREF rises to high; when the value of the core counter is less than the compare capture register (for example, the core counter increases to R16_TIM2_ATRLR value returns to all 0), OCxREF falls low.

10.3.6 One-pulse Mode

Single pulse mode generates a pulse in response to a specific event followed by a delay with programmable delay and pulse width. Setting the OPM bit can cause the core counter to stop when the next update event UEV is generated (the counter rolls over to 0).

Figure 10-4 Event generation and impulse response



As shown in Figure 10-4, a rising edge needs to be detected on the TI2 input pin. After a delay of Tdelay, a positive pulse of length Tpulse is generated on OC1:

1) Set TI2 to trigger. Setting the CC2S field to 01b to map TI2FP2 to TI2; setting the CC2P bit to 0b to set TI2FP2 as rising edge detection; setting the TS field to 110b to set TI2FP2 as trigger source; setting the SMS field to 110b to set TI2FP2 to be used to start the counter.
2) Tdelay is defined by the Compare Capture Register, Tpulse is determined by the value of the Auto Reload Value Register and the Compare Capture Register.

10.3.7 Synchronization of Timers

The timer can output clock pulses (TRGO) and can also receive input from other timers (ITRx). The sources of ITRx of different timers (TRGO of other timers) are different. The internal trigger connection of the timer is shown in Table 10-2.

Table 10-1 GTPM internal trigger connections

Slave timer	ITR0(TS=000)	ITR1(TS=001)	ITR2(TS=010)	ITR3(TS=011)
TIM2	TIM1			

10.3.8 Debug mode

When the system enters the debug mode, the timer can be controlled to continue running or stop according to the setting of DBG module.

10.4 Register Description

Table 10-2 TIM2-related registers list

Name	Access address	Description	Reset value
R16_TIM2_CTLR1	0x40000000	TIM2 control register 1	0x0000
R16_TIM2_SMCFGR	0x40000008	TIM2 slave mode control register	0x0000
R16_TIM2_DMAINTENR	0x4000000C	TIM2 interrupt enable register	0x0000
R16_TIM2_INTFR	0x40000010	TIM2 interrupt status register	0x0000
R16_TIM2_SWEVGR	0x40000014	TIM2 event generation register	0x0000
R16_TIM2_CHCTLR1	0x40000018	TIM2 compare/capture control register1	0x0000
R16_TIM2_CCER	0x40000020	TIM2 compare/capture enable register	0x0000
R16_TIM2_CNT	0x40000024	TIM2 counter	0x0000
R16_TIM2_PSC	0x40000028	TIM2 count clock prescaler	0x0000
R16_TIM2_ATRLR	0x4000002C	TIM2 auto-reload value register	0xFFFF
R32_TIM2_CH1CVR	0x40000034	TIM2 compare/capture register1	0x00000000
R32_TIM2_CH2CVR	0x40000038	TIM2 compare/capture register2	0x00000000

10.4.1 Control Register 1 (TIM2_CTLR1)

Offset address: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CAPLVL	CAPOV	Reserved		CO2NE	CO1NE	CKD[1:0]		ARPE	Reserved			OPM	URS	UDIS	CEN

Bit	Name	Access	Description	Reset value
15	CAPLVL	RW	In double-edge capture mode, the capture level indication is enabled. 0: Turn off the indication function; 1: Enable the indication function. *Note: After enabling, [16] in CHxCVR indicates the level corresponding to the captured value.*	0
14	CAPOV	RW	Capture value mode configuration. 0: The captured value is the actual counter value; 1: When the counter overflow occurs before capturing, the CHxCVR value is 0xFFFF.	0
[13:12]	Reserved	RO	Reserved	0
11	CO2NE	RW	Complementary output function - channel 2 and complementary channel output enable bit: 0: The dead zone function of channel 2 is turned off. 1: The dead zone function of channel 2 is enabled.	0
10	CO1NE	RW	Complementary output function - channel 1 and complementary channel output enable bit: 0: The dead zone function of channel 1 is turned off. 1: The dead zone function of channel 1 is enabled.	0
[9:8]	CKD[1:0]	RW	These 2 bits define the division ratio between the timer clock (CK_INT) frequency and the sampling clock used by the digital filter.	0

			00: Tdts=Tck_int; 01: Tdts= 2xTck_int; 10: Tdts= 4xTck_int; 11: Reserved.	
7	ARPE	RW	Auto-reinstall preinstallation enable bit. 0: Disable automatic reloading of value register (ATRLR); 1: Enable automatic reloading of value register (ATRLR).	0
[6:4]	Reserved	RO	Reserved	0
3	OPM	RW	One pulse mode. 0: The counter does not stop when the next update event occurs; 1: The counter stops when the next update event occurs (clearing the CEN bit).	0
2	URS	RW	Update request source, software selects the source of UEV events through this bit. 0: If the update interrupt request is enabled, any of the following events generates an update interrupt request: - Counter overflow/underflow - Set UG bit - Updates generated from the pattern controller 1: If the update interrupt request is enabled, only the counter overflow/underflow will generate an update interrupt request.	0
1	UDIS	RW	Disable updates. The software allows/disables the generation of UEV events through this bit. 0: UEV is allowed. Update (UEV) events are generated by any of the following events: -Set UG bit - Updates generated from the pattern controller Registers with cache are loaded with their preloaded values. 1: Disable UEV. No update event is generated and the registers (ATRLR, PSC, CHCTLRx) retain their values. If the UG bit is set or a hardware reset is issued from the mode controller, the counter and prescaler are reinitialized.	0
0	CEN	RW	Enable counter (Counterenable). 0: Disable counter. 1: Enable counter; *Note: External clock and gate control mode can work only after the CEN bit is set by software. Trigger mode automatically sets the CEN bit through hardware.*	0

10.4.2 Slave Mode Control Register (TIM2_SMCFGR)

Offset address: 0x08

15 14 13 12	11 10 9 8	7	6 5 4	3	2 1 0
DT2	DT1	Reserved	TS[2:0]	Reserved	SMS[2:0]

Bit	Name	Access	Description	Reset value
[15:12]	DT2	RW	Channel 2 dead time setting: T is the TIM2 module clock. 0000: The dead time is 1*T; 0001: The dead time is 2*T;	0000

			… 1110: The dead time is 15*T; 1111: The dead time is 16*T. *Note: The dead-time width must be smaller than the effective level width.*	
[11:8]	DT1	RW	Channel 1 dead time setting: T is the TIM2 module clock. 0000: The dead time is 1*T; 0001: The dead time is 2*T; … 1110: The dead time is 15*T; 1111: The dead time is 16*T. *Note: The dead-time width must be smaller than the effective level width.*	0000
7	Reserved	RO	Reserved	0
[6:4]	TS[2:0]	RW	Trigger selection field, these 3 bits select the trigger input source for the synchronized counter. 000: Internal trigger 0(ITR0); 001: Internal trigger 1(ITR1); 010: Internal trigger 2(ITR2); 011: Internal trigger 3(ITR3); 100: TI1Edge Detector (TI1F_ED); 101: Filtered timer input 1(TI1FP1); 110: Filtered timer input 2(TI2FP2); 111: Reserved; The above only changes when SMS is 0.	0
3	Reserved	RO	Reserved	0
[2:0]	SMS[2:0]	RW	Input mode selection field. Select the clock and trigger modes for the core counter. 000: Driven by the internal clock CK_INT. 001: Reserved; 010: Reserved; 011: Reserved; 100: Reset mode, where the rising edge of the trigger input (TRGI) will initialize the counter and generate a signal to update the registers. 101: Gated mode, when the trigger input (TRGI) is high, the counter clock is turned on; at the trigger input becomes low, the counter is stopped, and the counter starts and stops are controlled. 110: Trigger mode, where the counter is started on the rising edge of the trigger input TRGI and only the start of the counter is controlled. 111: External clock mode 1, rising edge of the selected trigger input (TRGI) drives the counter.	0

10.4.3 DMA/Interrupt Enable Register (TIM2_DMAINTENR)

Offset address: 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									TIE	Reserved			CC2IE	CC1IE	UIE

Bit	Name	Access	Description	Reset value
[15:7]	Reserved	RO	Reserved	0
6	TIE	RW	Trigger interrupt enable bit. 1: Enable trigger interrupt;	0

			0: Disable trigger interrupt.	
[5:3]	Reserved	RO	Reserved	0
2	CC2IE	RW	Compare capture channel 2 interrupt enable bit. 1: Enable compare capture channel 2 interrupt; 0: Disable compare capture channel 2 interrupt.	0
1	CC1IE	RW	Compare capture channel 1 interrupt enable bit. 1: Enable compare capture channel 1 interrupt; 0: Disable compare capture channel 1 interrupt.	0
0	UIE	RW	Update interrupt enable bit. 1: Enable update interrupt; 1: Disable update interrupt.	0

10.4.4 Interrupt Status Register (TIM2_INTFR)

Offset address: 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					CC2OF	CC1OF	Reserved		TIF	Reserved			CC2IF	CC1IF	UIF

Bit	Name	Access	Description	Reset value
[15:11]	Reserved	RO	Reserved	0
10	CC2OF	RW0	Compare capture channel 2 over-capture flag.	0
9	CC1OF	RW0	Compare capture channel 1 over-capture flag bit is only used when the compare capture channel is configured for input capture mode. This flag is set by hardware and a software write of 0 clears this bit. 1: The state of CC1IF has been set when the counter value is captured to the capture compare register; 0: No over-capture generated.	0
[8:7]	Reserved	RO	Reserved	0
6	TIF	RW0	The trigger interrupt flag bit, which is set by hardware to this position when a trigger event occurs and cleared by software. Trigger events include the detection of a valid edge at the TRGI input when from a mode other than gated, or any edge in gated mode. 1: Trigger event generated; 0: No trigger event generated.	0
[5:3]	Reserved	RO	Reserved	0
2	CC2IF	RW0	Compare capture channel 2 interrupt flag	0
1	CC1IF	RW0	Compare capture channel 1 interrupt flag If the compare capture channel is configured in output mode, this bit is set by hardware when the counter value matches the compare value, except in center-symmetric mode. This bit is cleared by software. 1: The value of the core counter matches the value of the compare capture register 1; 0: No match occurs. If compare capture channel 1 is configured in input mode, this bit is set by hardware when a capture event occurs, and it is cleared by software or by reading the compare capture register. 1: The counter value has been captured by comparing the capture register 1; 0: No input capture is generated.	0
0	UIF	RW0	Update interrupt flag bit. When an update event occurs, this bit is set by hardware and cleared by software.	0

			1: Update interrupt occurs; 0: No update event occurs. The following situations will generate update events: If UDIS=0, when the repetition counter value overflows or underflows; If URS=0, UDIS=0, when the UG bit is set, or when the counter core counter is reinitialized through software; If URS=0, UDIS=0, when the counter CNT is reinitialized by a trigger event;	

10.4.5 Event Generation Register (TIM2_SWEVGR)

Offset address: 0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									TG	Reserved			CC2G	CC1G	UG

Bit	Name	Access	Description	Reset value
[15:7]	Reserved	RO	Reserved	0
6	TG	WO	Trigger event generation bit, this bit is set by software and cleared by hardware to generate a trigger event. 1: A trigger event is generated, TIF is set, and if the corresponding interrupt is enabled, the corresponding interrupt is generated; 0: No action.	0
[5:3]	Reserved	RO	Reserved	0
2	CC2G	WO	Compare capture event generates bit 2. Generates compare capture event 2.	0
1	CC1G	WO	Compare capture event generates bit 1, generates compare capture event 1. This bit is set by software and cleared by hardware. Used to generate a compare capture event. 1: Generate a compare capture event on compare capture channel 1: If compare capture channel 1 is configured as output: set the CC1IF bit. If the corresponding interrupt is enabled, the corresponding interrupt is generated; If the comparison capture channel 1 is configured as input: the value of the current core counter is captured to the comparison capture register 1; the CC1IF bit is set, and if the corresponding interrupt is enabled, the corresponding interrupt is generated. of interruption. If CC1IF is already set, set CC1OF bit. 0: No action.	0
0	UG	WO	The update event generation bit generates an update event. This bit is set by software and automatically cleared by hardware. 1: Initialize the counter and generate an update event; 0: No action. *Note: The prescaler counter is also cleared, but the prescaler coefficient remains unchanged. If it is in the center symmetry mode or the up-counting mode, the core counter is cleared; if it is in the down-counting mode, the core counter takes the value of the reload value register.*	0

10.4.6 Compare/Capture Control Register 1 (TIM2_CHCTLR1)

Offset address: 0x18

Channels can be used as input (capture mode) or output (compare mode), and the direction of the channel is defined by the corresponding CCxS bit. The functions of other bits in this register are different in input and output modes. OCxx describes the function of the channel in output mode, and ICxx describes the function of the channel in input mode.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		Reserved	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			

Compare mode (pin direction is output).

Bit	Name	Access	Description	Reset value
15	Reserved	RO	Reserved	0
[14:12]	OC2M[2:0]	RW	Compare capture channel 2 mode setting fields. The 3 bits define the action of the output reference signal OC2REF, which determines the values of OC2, OC2N. OC2REF is active high, while the active levels of OC2 and OC2N depend on the CC2P, CC2NP bits. 000: Freeze. Comparison of the value of the capture register with the value of the comparison between the core counters does not work for OC2REF. 001: Force to set to valid level. Forcing OC2REF high when the core counter has the same value as the comparison capture register 2. 010: Force to set to invalid level. Forcing OC2REF low when the value of the core counter is the same as the comparison capture register 2. 011: Flip. Flips the level of OC2REF when the core counter is the same as the value of compare capture register 2. 100: Forced to invalid level. Forces OC1REF to low. 101: Force to valid level. Force OC1REF to high. 110: PWM mode 1: When counting up, once the core counter is less than the value of the comparison capture register, channel 2 is an effective level, otherwise it is an invalid level; 111: PWM mode 2: When counting up, once the core counter is less than the value of the comparison capture register, channel 2 is at an invalid level, otherwise it is at an effective level. *Note: Once the LOCK level is set to 3 and CC2S=00b, this bit cannot be modified. In PWM mode 1 or PWM mode 2, the OC2REF level changes only when the comparison result changes or when switching from freeze mode to PWM mode in output compare mode.*	0
11	OC2PE	RW	Compare capture register 2 preload enable bit. 1: Enable the preload function of compare capture register 2. Read and write operations only operate on the preload register. The preload value of compare capture register 2 is loaded into the current shadow register when the update event arrives; 0: Disable the preload function of compare capture	0

			register 2. Compare capture register 2 can be written to at any time, and the newly written value will take effect immediately. *Note: Once the LOCK level is set to 3 and CC2S=00, this bit cannot be modified. PWM mode can be used only in single pulse mode (OPM=1) without confirming the preload register, otherwise its action is uncertain.*	
10	OC2FE	RW	Compare capture channel 2 fast enable bit, which is used to speed up the response of the compare capture channel output to trigger input events. 1: The active edge of the input to the flipflop acts as if a comparison match has occurred. Therefore, the OC is set to the comparison level independent of the comparison result. The delay between the valid edge of the sample trigger and the output of the compare capture channel 2 is reduced to 3 clock cycles. 0: Based on the value of the counter and compare capture register 1, compare capture channel 2 operates normally, even if the flip-flop is open. The minimum delay to activate the compare capture channel 2 output is 5 clock cycles when the input of the flipflop has a valid edge. OC2FE only works when the channel is configured to PWM1 or PWM2 mode.	0
[9:8]	CC2S[1:0]	RW	Compare capture channel 2 input selection fields. 00: comparison capture channel 2 is configured as an output. 01: comparison capture channel 2 is configured as an input and IC2 is mapped on TI2. 10: comparison capture channel 2 is configured as an input and IC2 is mapped onTI1. 11: Compare Capture Channel 2 is configured as an input and IC2 is mapped on TRC. This mode works only when the internal trigger input is selected (by the TS bit). *Note: Compare capture channel 2 is writable only when the channel is off (when CC2E is zero).*	0
7	Reserved	RO	Reserved	0
[6:4]	OC1M[2:0]	RW	Compare capture channel 1 mode setting field.	0
3	OC1PE	RW	Compare capture register 1 preload enable bit.	0
2	OC1FE	RW	Compare capture channel 1 fast enable bit.	0
[1:0]	CC1S[1:0]	RW	Compare capture channel 1 input selection fields.	0

Capture mode (pin direction is input).

Bit	Name	Access	Description	Reset value
[15:12]	IC2F[3:0]	RW	Input capture filter 2 configuration field, these bits set the sampling frequency and digital filter length of the TI1 input. The digital filter consists of an event counter, which will generate an output transition after recording N events. 0000: No filter, sampled in fDTS; 1000: Sampling frequency Fsampling=Fdts/8, N=6; 0001: Sampling frequency Fsampling=Fck_int, N=2; 1001: Sampling frequency Fsampling=Fdts/8, N=8; 0010: Sampling frequency Fsampling=Fck_int, N=4; 1010: Sampling frequency Fsampling=Fdts/16, N=5;	0

			0011: Sampling frequency Fsampling=f=Fck_int, N=8; 1011: Sampling frequency Fsampling=Fdts/16, N=6; 0100: Sampling frequency Fsampling=Fdts/2, N=6; 1100: Sampling frequency Fsampling=Fdts/16, N=8; 0101: Sampling frequency Fsampling=Fdts/2, N=8; 1101: Sampling frequency Fsampling=Fdts/32, N=5; 0110: Sampling frequency Fsampling=Fdts/4, N=6; 1110: Sampling frequency Fsampling=Fdts/32, N=6; 0111: Sampling frequency Fsampling=Fdts/4, N=8; 1111: Sampling frequency Fsampling=Fdts/32, N=8.	
[11:10]	IC2PSC[1:0]	RW	Compare capture channel 2 prescaler configuration field, these 2 bits define the prescaler coefficient of compare capture channel 2. Once CC1E=0, the prescaler is reset. 00: No prescaler, each edge detected on the capture input port triggers a capture; 01: A capture is triggered every 2 events; 10: A capture is triggered every 4 events; 11: A capture is triggered every 8 events.	0
[9:8]	CC2S[1:0]	RW	Compare capture channel 2 input selection field. These 2 bits define the direction of the channel (input/output) and the selection of the input pin. 00: Compare capture channel 1 is configured as an output; 01: Compare capture channel 1 is configured as an input, and IC1 is mapped on TI1; 10: Compare capture channel 1 is configured as an input, and IC1 is mapped on TI2; 11: Compare capture channel 1 channel is configured as input, IC1 is mapped on TRC. This mode only works when the internal flip-flop input is selected (selected by the TS bit). *Note: CC1S is writable only when the channel is closed (CC1E is 0).*	0
[7:4]	IC1F[3:0]	RW	Input capture filter 1 configuration field.	0
[3:2]	IC1PSC[1:0]	RW	Compare capture channel 1 prescaler configuration field.	0
[1:0]	CC1S[1:0]	RW	Compare capture channel 1 input selection fields.	0

10.4.7 Compare/Capture Enable Register (TIM2_CCER)

Offset address: 0x20

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CO2NP	CO1NP	CO2P	CO1P	CC2P	CC2E	Reserved		CC1P	CC1E

Bit	Name	Access	Description	Reset value
[15:10]	Reserved	RO	Reserved	0
9	CO2NP	RW	Complementary output function - Channel 2 complementary channel output polarity setting bit: 0: High level active. 1: Low level active.	0
8	CO1NP	RW	Complementary output function - Channel 1 complementary channel output polarity setting bit: 0: High level active. 1: Low level active.	0

7	CO2P	RW	Complementary output function - Channel 2 output polarity setting bit: 0: High level active. 1: Low level active.	0
6	CO1P	RW	Complementary output function - Channel 1 output polarity setting bit: 0: High level active. 1: Low level active.	0
5	CC2P	RW	Compare capture channel 2 output polarity setting bit.	0
4	CC2E	RW	Compare capture channel 2 output enable bit.	0
[3:2]	Reserved	RO	Reserved	0
1	CC1P	RW	Compare capture channel 1 output polarity setting bit. CC1 channel is configured as output: 1: OC1 is active at low level; 0: OC1 is active at high level. CC1 channel configured as input: This bit selects whether IC1 or the inverted signal of IC1 is used as the trigger or capture signal. 1: Invert: Capture occurs on the falling edge of IC1; when used as an external trigger, IC1 is inverted. 0: No inversion: Capture occurs on the rising edge of IC1; when used as an external trigger, IC1 does not invert.	0
0	CC1E	RW	Compare capture channel 1 output enable bit. CC1 channel is configured as output: 1: On: OC1 signal is output to the corresponding output pin. 0: Off: OC1 output is disabled. CC1 channel is configured as input: This bit determines whether the counter value can be captured into the TIM2_CCR1 register. 1: Capture enabled; 0: Capture disabled.	0

10.4.8 Counter for General-purpose Timer (TIM2_CNT)

Offset address: 0x24

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	CNT[15:0]	RW	The real-time value of the timer's counter.	0

10.4.9 Counting Clock Prescaler (TIM2_PSC)

Offset address: 0x28

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	PSC[15:0]	RW	The division coefficient of the timer's prescaler; the clock frequency of the counter is equal to the input frequency of the divider/(PSC+1).	0

10.4.10 Auto-reload Value Register (TIM2_ATRLR)

Offset address: 0x2C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ATRLR[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	ATRLR[15:0]	RW	The value of ATRLR[15:0] will be loaded into the counter, read section 10.2.4 for when ATRLR acts and updates; the counter stops when ATRLR is empty.	0xFFFF

10.4.11 Compare/Capture Register 1 (TIM2_CH1CVR)

Offset address: 0x34

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL 1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH1CVR[15:0]															

Bit	Name	Access	Description	Reset value
[31:17]	Reserved	RO	Reserved	0
16	LEVEL1	RO	The level indication bit corresponding to the captured value	0
[15:0]	CH1CVR[15:0]	RW	The value of compare capture register channel 1.	0

10.4.12 Compare/Capture Register 2 (TIM2_CH2CVR)

Offset address: 0x38

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															LEVEL 2

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH2CVR[15:0]															

Bit	Name	Access	Description	Reset value
[31:17]	Reserved	RO	Reserved	0
16	LEVEL2	RO	The level indication bit corresponding to the captured value	0
[15:0]	CH2CVR[15:0]	RW	The value of compare capture register channel 2.	0

Chapter 11 Universal Synchronous Asynchronous Receiver Transmitter (USART)

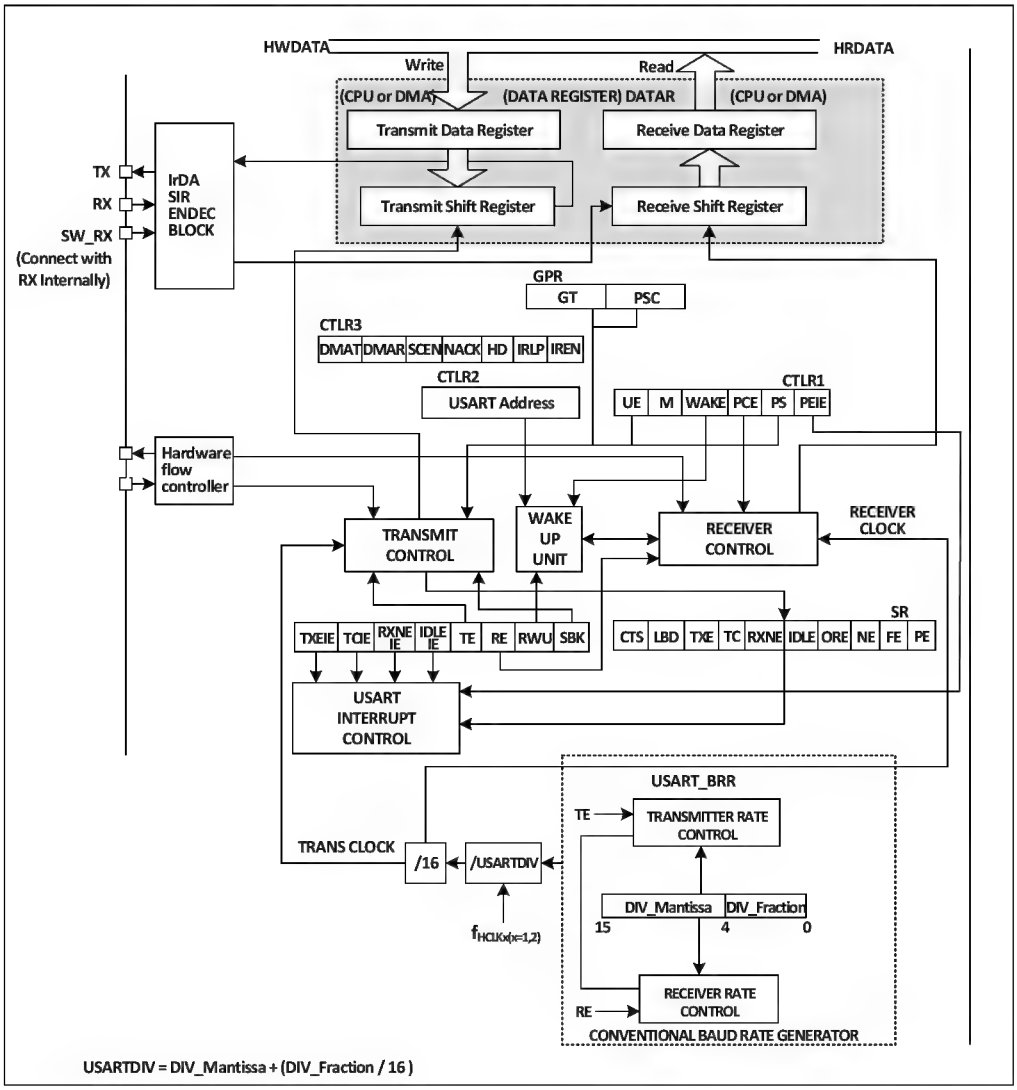
The module contains one Universal Synchronous Asynchronous Transceiver (USART).

11.1 Main Features

- Full-duplex or half-duplex synchronous or asynchronous communication
- NRZ data format
- Fractional baud rate generator, up to 3Mbps
- Programmable data length
- Configurable stop bits
- Support LIN, IrDA encoders
- Support DMA
- Multiple interrupt sources

11.2 Overview

Figure 11-1 Block diagram of a general-purpose asynchronous transceiver



When TE (transmit enable bit) is set, the data in the transmit shift register is output on the TX pin and the clock is output on the CK pin. When transmitting, the first bit shifted out is the least significant bit and each data frame starts with a low start bit, then the transmitter sends an 8- or 9-bit data word depending on the setting on the M (word length) bit, and finally a configurable number of stop bits. If equipped with a parity check bit, the last bit of the data word is the check bit. After the TE is set an idle frame is sent, which is 10 or 11 bits high and contains the stop bit. The disconnect frame is 10 or 11 bits low followed by the stop bit.

11.3 Baud Rate Generator

The baud rate of the transceiver = HCLK/(16*USARTDIV), HCLK is the clock of HB. The value of USARTDIV is determined by the two fields DIV_M and DIV_F in USART_BRR, which is calculated by the formula. The formula is as follows.

$$USARTDIV = DIV_M+(DIV_F/16)$$

It should be noted that the baud rate generated by the baud rate generator may not be just what the user needs, which

may be biased. In addition to taking the value as close as possible, a way to reduce the deviation can also be to increase the HB clock. For example, if you set the baud rate to 115200bps, the value of USARTDIV is set to 39.0625, which will give you a baud rate of exactly 115200bps at the highest frequency, but if you need a baud rate of 921600bps, the calculated USARTDIV is 4.88, but the closest value filled in USART_BRR is actually only 4.875, the actual baud rate is 923076bps, which is 0.16% error.

When the serial port waveform sent by the sender is transmitted to the receiver, there is a certain error in the baud rate of the receiver and the sender. The error mainly comes from three aspects: the actual baud rate of the receiver and the sender is inconsistent; there is an error in the clock of the receiver and the sender; and the change of the waveform in the line. The receiver of the peripheral module has a certain receiving tolerance. When the sum of the total deviations generated by the above three aspects is less than the tolerance limit of the module, this total deviation does not affect the transmission and reception. The tolerance limit of the module is affected by whether the fractional baud rate and M bits (data field word length) are used. The use of fractional baud rate and the use of 9-bit data field length will reduce the tolerance limit, but not less than 3%.

11.4 1-Wire Half-Duplex Mode

Half-duplex mode supports the use of a single pin (only the TX pin) for receiving and transmitting. The TX pin and RX pin are connected internally in the chip.

The way to turn on the half-duplex mode is to set the HDSEL bit in the control register 3 (R32_USARTx_CTLR3), but at the same time, you need to turn off the LIN mode and the infrared mode, that is, ensure that the IREN bit is in the reset state, which is in the control register 3 (R32_USART_CTLR3).

After setting to half-duplex mode, you need to set the TX IO port to multiplexed output high mode. With TE set, as long as data is written to the data register, it will be sent. In particular, it is important to note that in half-duplex mode, bus conflicts may occur when multiple devices use a single bus to send and receive, which requires users to use software to avoid it.

11.5 IrDA

The USART module supports control of IrDA infrared transceivers for physical layer communication. The LINEN, STOP, CLKEN, SCEN and HDSEL bits must be cleared to use IrDA. NRZ (non-return to zero) coding is used between the USART module and the SIR physical layer (infrared transceiver) and is supported up to 115200 bps rates.

IrDA is a half-duplex protocol, if UASRT is sending data to SIR physical layer, then IrDA decoder will ignore the newly sent IR signal, if USART is receiving data from SIR, then SIR will not accept the signal from USART. the level logic of USART to SIR and SIR to USART is different. In SIR receive logic, the high level is 1 and the low level is 0, but in SIR send logic, the high level is 0 and the low level is 1.

11.6 DMA

The USART module supports DMA function, which can be used to achieve fast and continuous transmitting and receiving. When DMA is enabled, the DMA writes data from the set memory space to the transmit buffer when TXE is set. When using DMA to receive, each time RXNE is set, DMA transfers the data in the receive buffer to a specific memory space.

11.7 Interrupt

The USART module supports a variety of interrupt sources, including transmit data register empty (TXE), CTS, transmit complete (TC), receive data ready (RXNE), data overflow (ORE), line idle (IDLE), parity error (PE), disconnect flag (LBD), noise (NE), overflow for multi-buffered communication (ORT), and frame error (FE), among others.

Table 11-1 Relationship between interrupts and corresponding enable bits

Interrupt source	Enable bit
Transmit data register empty (TXE)	TXEIE
CTS interrupt (CTS)	CTSIE
Transmission complete (TC)	TCIE
Receive data register not empty (data ready to be read)	RXNEIE
Overrun error detected (ORE)	
Idle line detected (IDLE)	IDLEIE
Parity error (PE)	PEIE
LIN break (LBD)	LBDIE
Noise flag (NE)	EIE
Overflow of multi-buffered communication (ORT)	
Frame error (FE) for multibuffered communication	

11.8 Register Description

Table 11-2 USART-related registers list

Name	Access address	Description	Reset value
R32_USART_STATR	0x40013800	USART status register	0x000000C0
R32_USART_DATAR	0x40013804	USART data register	0x00000000
R32_USART_BRR	0x40013808	USART baud rate	0x00000000
R32_USART_CTLR1	0x4001380C	USART control register 1	0x00000000
R32_USART_CTLR2	0x40013810	USART control register 2	0x00000000
R32_USART_CTLR3	0x40013814	USART control register 3	0x00000000
R32_USART_GPR	0x40013818	USART guard time and prescaler register	0x00000000

11.8.1 USART Status Register (USART_STATR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CTS	LBD	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE

Bit	Name	Access	Description	Reset value
[31:10]	Reserved	RO	Reserved	0
9	CTS	RW0	CTS state change flag. If the CTSE bit is set, this bit will be set high by hardware when the nCTS output state changes. It is cleared to zero by	0

			software. If the CTSIE bit is already set, an interrupt will be generated. 1: There is a change on the nCTS status line; 0: There is no change on the nCTS status line.	
8	LBD	RW0	LIN Break detection flag. This bit is set by hardware when LIN Break is detected. Cleared by software. If LBDIE has been set, an interrupt will be generated. 1: LIN Break is detected; 0: LIN Break is not detected.	0
7	TXE	RO	Transmit data register empty flag. This bit is set by hardware when the data in the TDR register is transferred to the shift register by hardware. If TXEIE has been set, an interrupt will be generated and the data register will be written and this bit will be reset. 1: The data has been transferred to the shift register; 0: The data has not been transferred to the shift register.	1
6	TC	RW0	Transmit completion flag. When a frame containing data is transmitted and TXE is set, the hardware will set this bit. If TCIE is set, a corresponding interrupt will be generated. The software will clear this bit after reading this bit and writing to the data register. You can also directly write 0 to clear this bit. 1: The transmitting is completed; 0: The transmitting is not yet completed.	1
5	RXNE	RW0	Read the data register non-empty flag. When the data in the shift register is transferred to the data register, this bit will be set by hardware. If RXNEIE has been set, the corresponding interrupt will also be generated. Reading the data register clears this bit. You can also write 0 directly to clear this bit. 1: The data has been received and can be read; 0: The data has not been received yet.	0
4	IDLE	RO	Bus idle flag. This bit will be set by hardware when the bus is free. If IDLEIE has been set, the corresponding interrupt will be generated. Reading the status register and then the data register clears this bit. 1: Bus is idle; 0: Bus idle is not detected. *Note: This bit will not be set again until RXNE is set.*	0
3	ORE	RO	Overload error flag. This bit will be set when there is data in the receiving shift register that needs to be transferred to the data register, but there is still data in the receiving field of the data register that has not been read out. If RXNEIE is set, a corresponding interrupt will be generated. 1: An overload error occurs; 0: There is no overload error. *Note: In case of an overload error, the value of the data register is not lost, but the value of the shift*	0

			register is overwritten. If the EIE able bit is set, the ORE flag position bit generates an interrupt in multi-buffer communication mode.	
2	NE	RO	Noise error flag. It is set by hardware when the noise error flag is detected. The operation of reading the status register and then reading the data register resets this bit. 1: Noise detected. 0: No noise is detected. *Note: This bit does not generate an interrupt. If the EIE bit is set, the FE flag position bit generates an interrupt in multi-buffer communication mode.*	0
1	FE	RO	Frame error flag. This bit will be set by hardware when a synchronization error, excessive noise or disconnect character is detected. Reading this bit and then reading the data register operation will reset this bit. 1: Frame error detected. 0: No frame error detected. *Note: This bit will not generate an interrupt. If the EIE bit is set, the FE flag position bit will generate an interrupt in multi-buffer communication mode.*	0
0	PE	RO	Check error flag. In receive mode, hardware sets this bit if a parity error occurs. Reading this bit and then reading the data register resets this bit. Software must wait for the RXNE flag to be set before clearing this bit. If PEIE has been set before, then setting this bit will generate a corresponding interrupt. 1: A parity error occurred; 0: No parity error.	0

11.8.2 USART Data Register (USART_DATAR)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							DR[8:0]								

Bit	Name	Access	Description	Reset value
[31:9]	Reserved	RO	Reserved	0
[8:0]	DR[8:0]	RW	data register. This register is actually composed of two registers: the receive data register (RDR) and the transmit data register (TDR). The read and write operations of DR start with reading the receive register (RDR) and writing the transmit register (TDR) respectively.	0

11.8.3 USART Baud Rate Register (USART_BRR)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIV_Mantissa[11:0]												DIV_Fraction[3:0]			

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
[15:4]	DIV_Mantissa[11:0]	RW	These 12 bits define the integer part of the dividing factor of the frequency divider.	0
[3:0]	DIV_Fraction[3:0]	RW	These 4 bits define the fractional part of the dividing factor of the frequency divider.	0

11.8.4 USART Control Register 1 (USART_CTLR1)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	RWU	SBK

Bit	Name	Access	Description	Reset value
[31:14]	Reserved	RO	Reserved	0
13	UE	RW	USART enable bit. When this bit is cleared, the USART divider and output will stop working after the current byte transmission is completed.	0
12	M	RW	Word length. 1: 9 data bits; 0: 8 data bits.	0
11	WAKE	RW	Wake-up bit. This bit determines the method to wake up the USART. 1: Address mark; 0: Bus is idle.	0
10	PCE	RW	Check bit enabled. For the receiver, it is to perform parity check on the data; for the sender, it is to insert the check bit. Once this bit is set, the check bit enable will not take effect until the current byte transmission is completed.	0
9	PS	RW	Parity selection. 0 means even parity, 1 means odd parity. After this bit is set, the check bit enable will take effect only after the current byte transmission is completed.	0
8	PEIE	RW	Parity check interrupt enable bit. Setting this bit indicates that a parity error interrupt is enabled.	0
7	TXEIE	RW	Transmit buffer empty interrupt enabled. Setting this bit indicates that the transmit buffer empty interrupt is allowed.	0
6	TCIE	RW	Transmit complete interrupt enabled. Setting this bit indicates that a transmission completion interrupt is allowed.	0
5	RXNEIE	RW	Receive buffer not empty interrupt enabled. Setting this bit indicates that the receive buffer is not empty interrupt is allowed.	0
4	IDLEIE	RW	Bus idle interrupt enabled. Setting this bit indicates that the bus idle interrupt is enabled.	0

3	TE	RW	Transmit enable. Setting this bit enables the transmitter.	0
2	RE	RW	Receive Enable. Setting this bit enables the receiver, which starts detecting the start bit on the RX pin.	0
1	RWU	RW	Receive wakeup. This bit determines whether to put the USART into silent mode. 1: The receiver is in silent mode; 0: The receiver is in normal working mode. *Note 1: Before setting the RWU bit, the USART needs to receive a data byte first, otherwise it cannot be woken up by the bus idle mode in silent mode;* *Note 2: When configured as address tag wake-up, the RWU bit cannot be modified by software when RXNE is set.*	0
0	SBK	RW	Transmit frame break character control bit. Set this bit to transmit a frame break character. Reset by hardware when the stop bit of the frame is broken. 1: Transmit; 0: Do not transmit.	0

11.8.5 USART Control Register 2 (USART_CTLR2)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	LINEN	STOP[1:0]		Reserved					LBDIE	LBDL	Reserved	ADD[3:0]			

Bit	Name	Access	Description	Reset value
[31:15]	Reserved	RO	Reserved	0
14	LINEN	RW	LIN mode enable bit, set to enable LIN mode. In LIN mode, the SBK bit can be used to send the LIN sync break symbol and detect the LIN sync break symbol.	0
[13:12]	STOP[1:0]	RW	Stop bit setting field. These two bits are used to set the stop bit. 00: 1 stop bit; 01: 0.5 stop bits; 10: 2 stop bits; 11: 1.5 stop bits.	0
[11:7]	Reserved	RO	Reserved	0
6	LBDIE	RW	LIN Break detects interrupt enable, and setting this bit will enable interrupts caused by LBD;	0
5	LBDL	RW	LIN Break detects the length. This bit is used to select whether it is 11-bit or 10-bit break detection. 1: 11-bit break detection; 0: 10-bit break detection.	0
4	Reserved	RW	Reserved	0
[3:0]	ADD[3:0]	RW	Address field, used to set the USART node address of this device. Used in silent mode under multi-processor communication, using address tags to wake up a USART device.	0

11.8.6 USART Control Register 3 (USART_CTLR3)

Offset address: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					CTSIE	CTSE	RTSE	DMAT	DMAR	Reserved		HDSEL	IRLP	IREN	EIE

Bit	Name	Access	Description	Reset value
[31:11]	Reserved	RO	Reserved	0
10	CTSIE	RW	CTSIE interrupt enable bit. When this bit is set, an interrupt will be generated when CTS is set.	0
9	CTSE	RW	CTS enable bit, setting this bit will enable CTS flow control.	0
8	RTSE	RW	RTS enable bit, setting this bit will enable RTS flow control.	0
7	DMAT	RW	DMA transmit enable bit. Set this bit to 1 to use DMA when transmitting.	0
6	DMAR	RW	DMA receive enable bit. Set this bit to 1 to use DMA when receiving.	0
[5:4]	Reserved	RO	Reserved	0
3	HDSEL	RW	Half-duplex mode selection bit, set this bit to select half-duplex mode.	0
2	IRLP	RW	Infrared low-power consumption selection bit, set this bit to enable low power consumption mode when infrared is selected.	0
1	IREN	RW	Infrared enable bit, set this bit to enable infrared mode.	0
0	EIE	RW	Error enable interrupt bit. After setting this bit, if FE, ORE or NE is set, an interrupt will be generated under the premise that DMAR is set.	0

11.8.7 USART Guard Time and Prescaler Register (USART_GPR)

Offset address: 0x18

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								PSC[7:0]							

Bit	Name	Access	Description	Reset value
[31:8]	Reserved	RO	Reserved	0
[7:0]	PSC[7:0]	RW	Prescaler value range. In the infrared low-power mode, the source clock is divided by this value (all 8 bits are valid). When the value is 0, it means reserved; in the infrared normal mode, this bit can only be set to 1.	0

Chapter 12 Inter-integrated Circuit (I2C) interface

The internal integrated circuit bus (I2C) is widely used for communication between microcontrollers, sensors and other off-chip modules. It supports multi-slave mode and can operate at 100KHz (standard) and 400KHz (fast) using only two wires (SDA and SCL). The I2C bus not only supports I2C timing, but also supports DMA function and CRC check function.

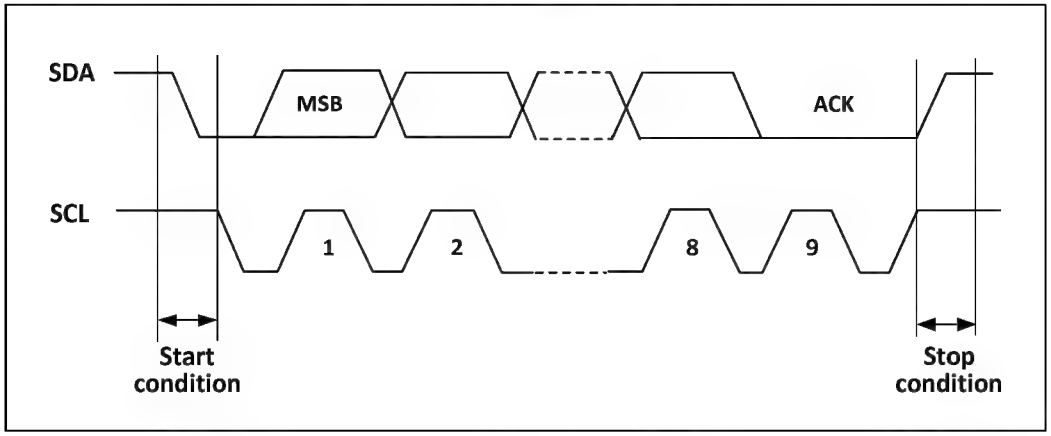
12.1 Main Features

- Support slave mode.
- Support 7-bit or 10-bit addresses.
- Slave devices support dual 7-bit addresses.
- Support two speed modes: 100KHz and 400KHz
- Various error flags
- Support extended clock function
- 2 interrupt vectors
- Support DMA, PEC

12.2 Overview

I2C is a half-duplex bus, which can only operate in one of the following two modes at the same time: slave sending mode and slave receiving mode. Both data and address are transmitted in units of 8 bits, with the high bit in front and the low bit in the back. After the start event is a byte (in 7-bit address mode) or two bytes (in 10-bit address mode) address, every time the master transmits 8 bits of data or address, the slave needs to reply with a response ACK, that is, pull the SDA bus low, as shown in Figure 12-1.

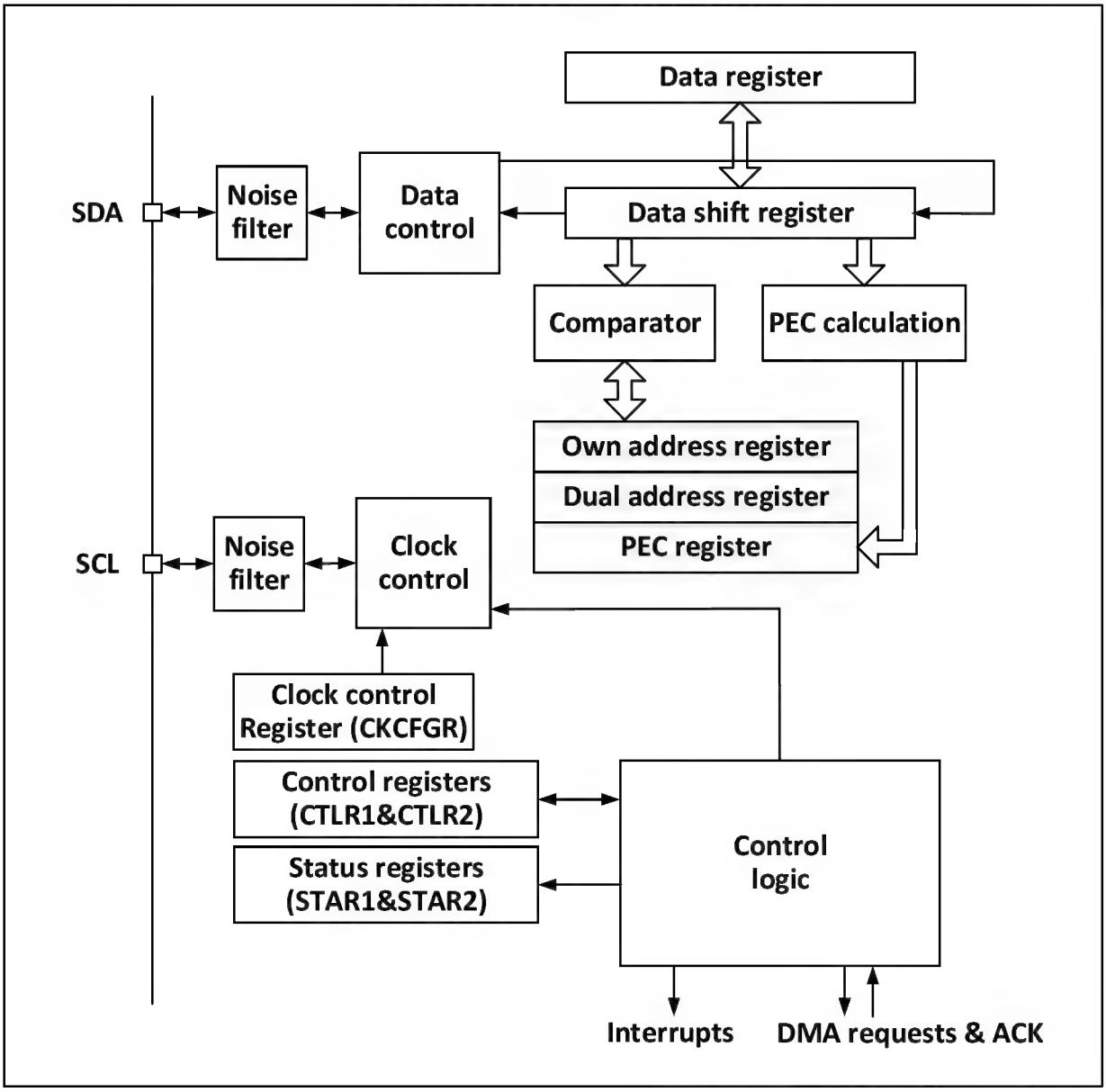
Figure 12-1 I2C timing diagram



For normal use, the correct clock must be input to I2C. In standard mode, the minimum input clock is 2MHz, and in fast mode, the minimum input clock is 4MHz.

Figure 12-2 shows the functional block diagram of the I2C module.

Figure 12-2 I2C functional block diagram



12.3 Slave Mode

In slave mode, the I2C module can recognize its own address and the general call address. The software can control whether to enable or disable the recognition of broadcast call addresses. Once a start event is detected, the I2C module compares the SDA data through the shift register with its own address (the number of bits depends on ENDUAL and ADDMODE) or the broadcast address (when ENGC is set). If there is no match, it will be ignored until generate a new start event; if it matches the header sequence, it will generate an ACK signal and wait for the address of the second byte; if the address of the second byte also matches or the full segment address matches in the case of a 7-bit address, So:

First generate an ACK response;

The ADDR bit is set, and if the ITEVTEN bit is set, a corresponding interrupt will be generated;

If you are using dual address mode (the ENDUAL bit is set), you also need to read the DUALF bit to determine which address is evoked by the host.

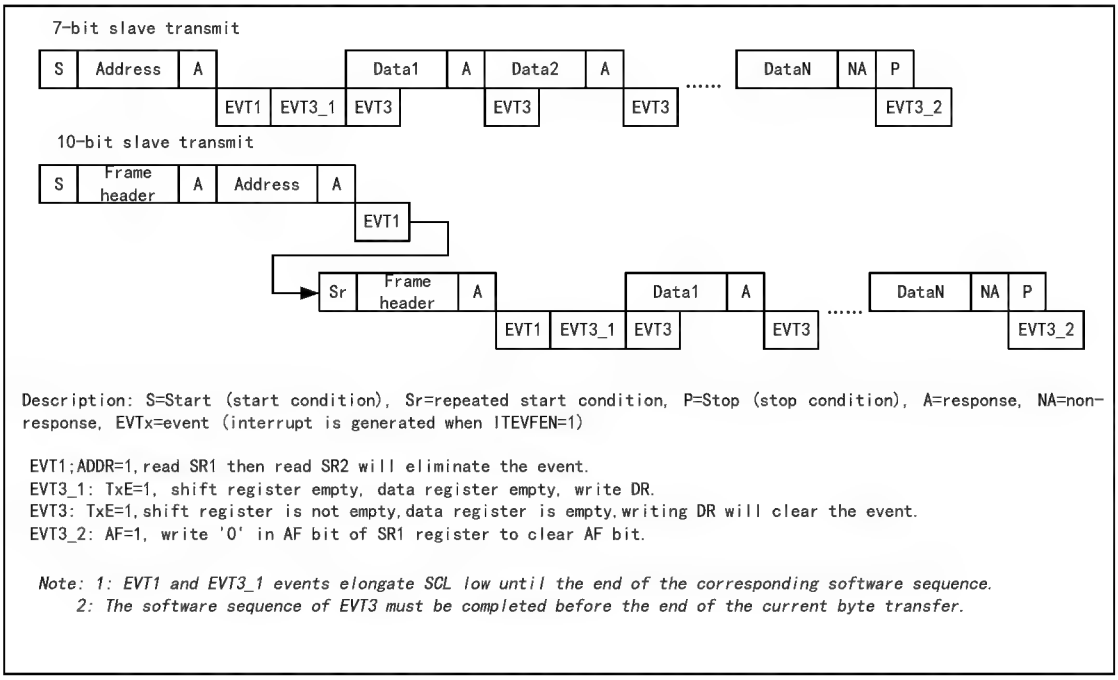
The default slave mode is the receive mode. After the last bit of the received header sequence is 1, or the last bit of

the 7-bit address is 1 (depending on whether the header sequence is received for the first time or an ordinary 7-bit address), when a repeated When the start condition is reached, the I2C module will enter the transmitter mode, and the TRA bit will indicate whether it is the receiver or transmitter mode.

Slave transmit mode:

After clearing the ADDR bit, the I2C module sends the byte from the data register through the shift register to the SDA line. The slave holds SCL low until the ADDR bit is cleared and the data to be transmitted has been written to the data register. (See EVT1 and EVT3 in the picture below). After receiving a response ACK, the TxE bit will be set, and if ITEVTEN and ITBUFEN are set, an interrupt will be generated. If TxE is set but no new data is written to the data register before the end of the next data transmission, the BTF bit will be set. Before clearing BTF, SCL will remain low. After reading status register 1 (R16_I2C_STAR1), writing data to the data register will clear the BTF bit.

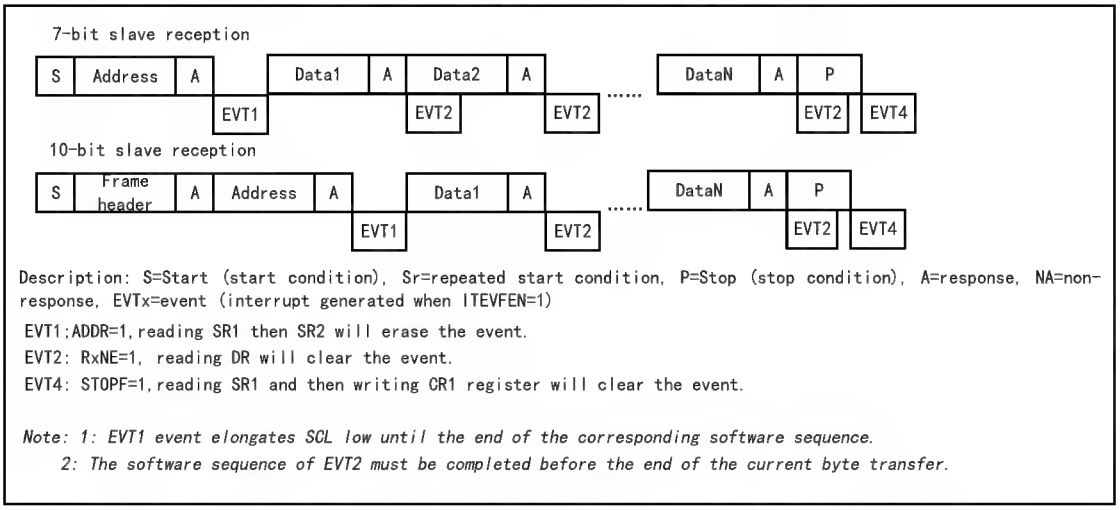
Figure 12-3 Slave transmitter transmission sequence diagram



Slave receive mode:

After ADDR is cleared, the I2C module stores the data on SDA into the data register through the shift register. After each byte is received, the I2C module sets an ACK bit and sets the RxNE bit. If ITEVTEN and ITBUFEN are set, an interrupt will also be generated. If RxNE is set and old data is not read before new data is received, BTF will be set. SCL will remain low until the BTF bit is cleared. Reading status register 1 (R16_I2C_STAR1) and reading the data in the data register will clear the BTF bit.

Figure 12-4 Receiver transmission sequence diagram



The master device will generate a stop condition after transmitting the last data byte. When the I2C module detects a stop event, the STOPF bit will be set. If the ITEVFEN bit is set, an interrupt will be generated. The user needs to read the status register (R16_I2C_STAR1) and then write the control register (such as the reset control word SWRST) to clear it. (See EVT4 in the picture above).

12.4 Error Conditions

12.4.1 Bus Error (BERR)

During the transfer of address or data, when the I2C module detects an external start or stop event, a bus error will be generated. When a bus error occurs, the BERR bit is set, and an interrupt is generated if ITERREN is set. In slave mode, data is discarded and the hardware releases the bus. If it is a start signal, the hardware will consider it a restart signal and start waiting for the address or stop signal; if it is a stop signal, it will operate according to normal stop conditions in advance.

12.4.2 Acknowledge Failure (AF)

When the I2C module detects a byte and does not respond, a response error occurs. When a response error occurs: AF will be set, and if ITERREN is set, an interrupt will be generated; when an AF error is encountered, the hardware must release the bus.

12.4.3 Overrun/Underrun Error (OVR)

- **Overrun error**

In slave mode, if clock extension is disabled and the I2C module is receiving data, if one byte of data has been received but the last received data has not been read out, an overload error will occur. When an overload error occurs, the last received byte will be discarded and the sender should resend the last byte sent.

- **Underrun error**

In slave mode, if clock stretching is disabled and the I2C module is sending data, if new data has not been written to the data register before the next byte clock arrives, an underrun error will occur. When an underrun error occurs, the data in the previous data register will be sent twice. If an underrun error occurs, the receiver should discard the duplicated data. In order not to generate an underrun error, the I2C module should write data to the data register

before the first rising edge of the next byte.

12.5 Clock Extension

If clock stretching is disabled, there is a possibility of overrun/underrun errors. But if clock stretching is enabled:

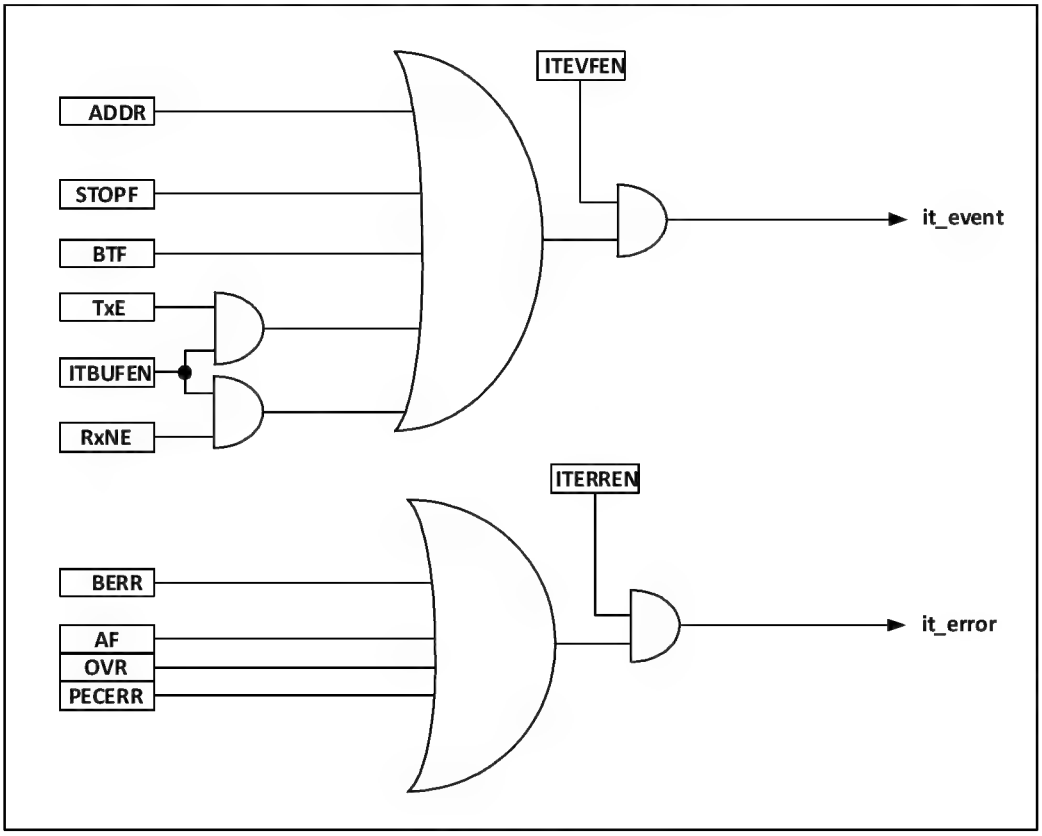
- In transmit mode, if TxE is set and BTF is set, SCL will always be low, waiting for the user to read the status register and write the data to be sent to the data register;
- In receive mode, if RxNE is set and BTF is set, SCL will remain low after receiving data until the user reads the status register and reads the data register.

It can be seen that enabling clock stretching can avoid overload/underrun errors.

12.6 Interrupts

Each I2C module has two interrupt vectors, namely event interrupt and error interrupt. Two interrupts support the interrupt sources in Figure 12-5.

Figure 12-5 I2C Interrupt Request



12.7 DMA

DMA can be used to transmit and receive batch data. The ITBUFEN bit of the control register cannot be set when using DMA.

- Transmit by DMA

DMA mode can be activated by setting the DMAEN bit in the CTLR2 register. As long as the TxE bit is set, data

will be loaded from the set memory into the I2C data register by DMA. The following settings are required to assign channels to I2C.

1) The I2C_DATAR register address is set to the DMA_PADDRx register and the memory address is set in the DMA_MADDRx register so that data will be sent from memory to the I2C_DATAR register after each TxE event.
2) Set the required number of bytes to transfer in the DMA_CNTRx register. After each TxE event, this value will be decremented.
3) Configure the channel priority using the PL[0:1] bits in the DMA_CFGRx register.
4) Set the DIR bit in the DMA_CFGRx register and configure an interrupt request to be issued when half or all of the entire transfer is completed, depending on the application requirements.
5) Activate the channel by setting the EN bit on the DMA_CFGRx register.

When the number of data transfer bytes set in the DMA controller has been completed, the DMA controller sends an end-of-transmission EOT/EOT_1 signal to the I2C interface. With interrupts enabled, a DMA interrupt will be generated.

- Receive by DMA

After setting DMAEN in the CTLR2 register, the DMA receive mode can be entered. When using DMA to receive, DMA transfers the data in the data register to the preset memory area. The following steps are required to assign channels to I2C.

1) Set the I2C_DATAR register address to the DMA_PADDRx register and the memory address in the DMA_MADDRx register so that data will be written from the I2C_DATAR register to the memory after each RxNE event.
2) Set the required number of bytes to transfer in the DMA_CNTRx register. After each RxNE event, this value will be decremented.
3) Configure the channel priority with PL[0:1] in the DMA_CFGRx register.
4) Clear the DIR bit in the DMA_CFGRx register. Depending on the application requirements, you can set an interrupt request to be issued when the data transfer is half or fully completed.
5) Setting the EN bit in the DMA_CFGRx register activates the channel.

When the number of data transfer bytes set in the DMA controller has been completed, the DMA controller sends an end-of-transmission EOT/EOT_1 signal to the I2C interface. With interrupts enabled, a DMA interrupt will be generated.

12.8 Packet Error Checking

Packet Error Checking (PEC) is a step that adds a CRC8 check to provide transmission reliability. It uses the following polynomial to calculate each bit of serial data:

$$C=X^8+X^2+X+1$$

The PEC calculation is activated by the ENPEC bit of the control register and is calculated on all information bytes, including address and read and write bits. When sending, enabling PEC will add a byte of CRC8 calculation result after the last byte of data; while in receiving mode, the last byte is considered to be the CRC8 check result. If it does not match the internal calculation result, a NAK will be replied. If it is the main receiver, a NAK will be replied regardless of whether the verification result is correct or not.

12.9 Register Description

Table 12-1 I2C-related registers list

Name	Access address	Description	Reset value
R16_I2C_CTLR1	0x40005400	I2C control register 1	0x0000
R16_I2C_CTLR2	0x40005404	I2C control register 2	0x0000
R16_I2C_OADDR1	0x40005408	I2C address register 1	0x0000
R16_I2C_OADDR2	0x4000540C	I2C address register 2	0x0000
R16_I2C_DATAR	0x40005410	I2C data register	0x0000
R16_I2C_STAR1	0x40005414	I2C status register 1	0x0000
R16_I2C_STAR2	0x40005418	I2C status register 2	0x0000

12.9.1 I2C Control Register (I2C_CTLR1)

Offset address: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWRST	Reserved		PEC	POS	ACK	Reserved		NOSTRETCH	ENGC	ENPEC	Reserved				PE

Bit	Name	Access	Description	Reset value
15	SWRST	RW	Software reset, user code setting this bit will reset the I2C peripheral. Before resetting, make sure that the pins of the I2C bus are released and the bus is in an idle state. *Note: This bit can reset the I2C module when no stop condition is detected on the bus but the busy bit is 1.*	0
[14:13]	Reserved	RO	Reserved	0
12	PEC	RW	Packet error detection enable bit, set this bit to enable packet error detection. User code can set or clear this bit; when the PEC is transmitted, a start or end signal is generated, or the PE bit is cleared to 0, the hardware clears this bit. 1: With PEC; 0: Without PEC.	0
11	POS	RW	The ACK and PEC position set bits can be set or cleared by user code. After PE is cleared, it can be cleared by hardware. 1: The ACK bit controls the ACK or NAK of the next byte received in the shift register. The next byte received in the PEC shift register is PEC; 0: The ACK bit controls the ACK or NAK of the byte currently being received in the shift register. The PEC bit indicates that the current byte in the shift register is PEC. *Note: The usage of POS bit in 2-byte data reception is as follows: it must be configured before receiving. In order to NACK the second byte, the ACK bit must be cleared immediately after clearing the ADDR bit; in order to detect the PEC of the second byte, the PEC bit must be set after the POS bit is configured after the ADDR event occurs.*	0

10	ACK	RW	Acknowledge enable bit. This bit can be set or cleared by user code. When the PE bit is set, this bit can be cleared by hardware. 1: Return a response after receiving a byte; 0: No response.	0
[9:8]	Reserved	RO	Reserved	0
7	NOSTRETCH	RW	Disable clock stretching bit. This bit is used to disable clock stretching in slave mode when the ADDR or BTF flag is set until cleared by software. 1: Disable clock extension; 0: Enable clock extension.	0
6	ENGC	RW	Broadcast call enable bit, set this bit to enable broadcast call and respond to broadcast address 00h.	0
5	ENPEC	RW	PEC enable bit, set this bit to enable PEC calculation.	0
[4:1]	Reserved	RO	Reserved	0
0	PE	RW	I2C peripheral enable bit. 1: Enable the I2C module; 0: Disable the I2C module.	0

12.9.2 I2C Control Register 2 (I2C_CTLR2)

Offset address: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				DMA EN	ITBU FEN	ITEV TEN	ITER REN	Reserved		FREQ[5:0]					

Bit	Name	Access	Description	Reset value
[15:12]	Reserved	RO	Reserved	0
11	DMAEN	RW	DMA request enable bit, set to allow DMA requests when TxE or RxEN is set.	0
10	ITBUFEN	RW	Buffer interrupt enable bit. 1: When TxE or RxEN is set, an event interrupt is generated; 0: No interrupt is generated when TxE or RxEN is set.	0
9	ITEVTEN	RW	Event interrupt enable bit, setting this bit enables event interrupt. This interrupt will be generated under the following conditions: ADDR=1 (slave mode); STOPF=1 (slave mode); BTF=1, but no TxE or RxEN events; If ITBUFEN=1, the TxE event is 1; A RxNE event of 1 if ITBUFEN=1.	0
8	ITERREN	RW	Error interrupt enable bit, set to enable error interrupt. This interrupt will be generated under the following conditions: BERR=1; AF=1; OVR=1; PECERR=1;	0
[7:6]	Reserved	RO	Reserved	0
[5:0]	FREQ[5:0]	RW	In the I2C module clock frequency domain, the correct clock frequency must be entered to generate the correct timing. The allowed range is between 8-48MHz. Must be set between 001000b	0

			and 110000b in MHz.	

12.9.3 I2C Address Register 1 (I2C_OADDR1)

Offset address: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDMODE	Reserved					ADD[9:8]		ADD[7:1]							ADD0

Bit	Name	Access	Description	Reset value
15	ADDMODE	RW	Address mode. 1: 10-bit slave address (does not respond to 7-bit address); 0: 7-bit slave address (does not respond to 10-bit address).	0
[14:10]	Reserved	RO	Reserved	0
[9:8]	ADD[9:8]	RW	Interface address, bits 9-8 when using a 10-bit address, ignored when using a 7-bit address.	0
[7:1]	ADD[7:1]	RW	Interface address, bits 7-1.	0
0	ADD0	RW	Interface address, bit 0 when using a 10-bit address, ignored when using a 7-bit address.	0

12.9.4 I2C Address Register 2 (I2C_OADDR2)

Offset address: 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								ADD2[7:1]							ENDUAL

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
[7:1]	ADD2[7:1]	RW	Interface address, bits 7-1 of the address in dual address mode.	0
0	ENDUAL	RW	Dual address mode enable bit, setting this bit allows ADD2 to be recognized.	0

12.9.5 I2C Data Register (I2C_DATAR)

Offset address: 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DR[7:0]							

Bit	Name	Access	Description	Reset value
15:8	Reserved	RO	Reserved	0
7:0	DR[7:0]	RW	Data register, this field is used to store received data or data for sending to the bus.	0

12.9.6 I2C Status Register 1 (I2C_STAR1)

Offset address: 0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			PECERR	OVR	AF	Reserved	BERR	TxE	RxNE	Reserved	STOPF	Reserved	BTF	ADDR	Reserved

Bit	Name	Access	Description	Reset value
[15:13]	Reserved	RO	Reserved	0
12	PECERR	RW0	The PEC error flag bit occurs during reception. This bit can be reset by the user writing 0, or reset by hardware when PE goes low. 1: There is a PEC error. After receiving the PEC, NAK is returned; 0: No PEC error.	0
11	OVR	RW0	Overload and underload flags. 1: An overload or underload event occurs: When NOSTRETCH=1, when a new byte is received in the receiving mode, the content in the data register has not yet been read, and the newly received byte will be lost; In transmit mode, no new data is written to the data register, and the same byte will be sent twice; 0: No overload or underload events.	0
10	AF	RW0	Acknowledgment failure flag bit, this bit can be reset by the user writing 0, or reset by hardware when PE goes low. 1: The response is wrong; 0: The response is normal.	0
9	Reserved	RO	Reserved	0
8	BERR	RW0	Bus error flag bit, this bit can be reset by the user writing 0, or reset by hardware when PE goes low. 1: There is an error in the start or stop condition; 0: Normal.	0
7	TxE	RO	The data register is an empty flag, which can be cleared by writing data to the data register, or is automatically cleared by hardware after a start or stop bit is generated, or when PE is 0. 1: When sending data, the sending data register is empty; 0: The data register is not empty.	0
6	RxNE	RO	The data register is not empty flag bit. Read and write operations on the data register will clear this bit, or when PE is 0, this bit will be cleared by hardware. 1: When receiving data, the data register is not empty; 0: Normal.	0
5	Reserved	RO	Reserved	0
4	STOPF	RO	Stop event flag bit. After the user reads status register 1, the write operation to control register 1 will clear this bit, or when PE is 0, this bit will be cleared by hardware. 1: After acknowledgment, the slave device detects a stop event on the bus; 0: No stop event is detected.	0

3	Reserved	RO	Reserved	0
2	BTF	RO	Byte transmission end flag bit. After the user reads status register 1, reading and writing the data register will clear this bit; during transmission, after a start or stop event is initiated, or when PE is 0, this bit will be cleared by hardware. 1: Byte transmission ends. When NOSTRETCH=0: When sending, when a new data is sent and the data register has not been written with new data; when receiving, when a new byte is received but the data register has not been read; 0: None.	0
1	ADDR	RW0	Address match flag bit. After the user reads status register 1, the read operation of status register 2 will clear this bit, or when PE is 0, this bit will be cleared by hardware. Slave mode: 1: The received address matches; 0: The address does not match or no address was received.	0
0	Reserved	RO	Reserved	0

12.9.7 I2C Status Register 2 (I2C_STAR2)

Offset address: 0x18

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PEC[7:0]								DUALF	Reserved		GENCALL	Reserved	TRA	BUSY	Reserved

Bit	Name	Access	Description	Reset value
[15:8]	PEC[7:0]	RO	Packet error check field. When PEC is enabled (ENPEC is set), this field stores the value of PEC.	0
7	DUALF	RO	Match detection flag bit, the hardware will clear this bit when a stop bit or start bit is generated, or when PE=0. 1: The received address matches the content in OAR2; 0: The received address matches the content in OAR1.	0
[6:5]	Reserved	RO	Reserved	0
4	GENCALL	RO	Broadcast call address flag bit, which is cleared by the hardware when a stop or start bit is generated, or when PE=0. 1: Broadcast call address received when ENGC=1; 0: Address at which no broadcast call was received.	0
3	Reserved	RO	Reserved	0
2	TRA	RO	The transmit/receive flag bit will be cleared by hardware when a stop event (STOPF=1), repeated start condition, and PE=0 are detected. 1: Data has been transmitted; 0: Data has been received. This bit is determined based on the R/W bit of the	0

			address byte.	
1	BUSY	RO	Bus busy flag, this bit is cleared when a stop bit is detected. This information is still updated when the interface is disabled (PE=0). 1: The bus is busy: SDA or SCL is at low level; 0: The bus is idle and there is no communication.	0
0	Reserved	RO	Reserved	0

Chapter 13 Electronic Signature (ESIG)

The electronic signature contains chip identification information: flash memory area capacity and unique identification. It is burned into the system storage area of the memory module by the manufacturer when leaving the factory, and can be read through SWD (SDI) or application code.

13.1 Function Description

Flash memory area capacity: Indicates the size that the current chip user application can use.
Unique identification: 96-bit binary code, unique to any microcontroller, users can only read and access it and cannot modify it. This unique identification information can be used as the security password, decryption key, product serial number, etc. of the microcontroller (product) to improve the system security mechanism or indicate identity information.
Users of the above content can perform read access in 8/16/32 bits.

13.2 Register Description

Table 13-1 ESIG registers

Name	Access address	Description	Reset value
R16_ESIG_FLACAP	0x1FFFF7E0	Flash memory capacity register	0xXXXX
R32_ESIG_UNIID1	0x1FFFF7E8	UID register 1	0xXXXXXXXX
R32_ESIG_UNIID2	0x1FFFF7EC	UID register 2	0xXXXXXXXX
R32_ESIG_UNIID3	0x1FFFF7F0	UID register 3	0xXXXXXXXX

13.2.1 Flash Capacity Register (ESIG_FLACAP)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
F_SIZE[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	F_SIZE[15:0]	RO	Flash memory capacity in Kbytes. Example: 0x0080 = 128 K bytes.	X

13.2.2 UID Register (ESIG_UNIID1)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	U_ID[31:0]	RO	UID bits 0-31.	X

13.2.3 UID Register (ESIG_UNIID2)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID[63:48]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID[47:32]															

Bit	Name	Access	Description	Reset value
[31:0]	U_ID[63:32]	RO	UID bits 32-63.	X

13.2.4 UID Register (ESIG_UNIID3)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID[95:80]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID[79:64]															

Bit	Name	Access	Description	Reset value
[31:0]	U_ID[95:64]	RO	UID bits 64-95.	X

Chapter 14 Flash Memory and User Option Bytes

14.1 Flash Memory Organization

The internal flash memory organization structure of the chip is as follows:

Table 14-1 Flash memory organization structure

Block	Name	Address Range	Size(byte)
Main memory	Page 0	0x0800 0000 – 0x0800 003F	64
	Page 1	0x0800 0040 – 0x0800 007F	64
	Page 2	0x0800 0080 – 0x0800 00BF	64
	Page 3	0x0800 00C0 – 0x0800 00FF	64
	…	…	…
	Page 256	0x0800 3FC0 – 0x0800 3FFF	64
Information block	Startup program code	0x1FFF F000 – 0x1FFF F77F	2K-128
	User option byte	0x1FFF F800 – 0x1FFF F83F	64

Note:

1) The above-mentioned main memory area is used for user application storage and is write-protected in units of 1K bytes (16 pages). Except for the "manufacturer configuration word" area, which is factory-locked and inaccessible to users, other areas can be operated by users under certain conditions.

14.2 Flash Memory Programming and Safety

14.2.1 Program/Erase Methods

- Standard programming: This method is the default programming method (compatibility mode). In this mode, the CPU performs programming in a single 2-byte manner, and performs erase and whole-chip erase operations in a single 1K byte manner.
- Fast programming: This method uses page operation mode (recommended). After a specific sequence is unlocked, a single 64-byte programming, 64-byte erase, and 1K-byte erase are performed (standard 1K full-chip erase is also suitable for fast programming).

14.2.2 Security - Protection against Illegal Access (read, write, erase)

- Page write protection
- Read protection

When the chip is in read protection state:

1) Main memory pages 0-32 (2K bytes) are automatically write-protected and are not controlled by the FLASH_WPR register; when the read-protected state is released, all main memory pages are controlled by the FLASH_WPR register.
2) The main memory cannot be erased or programmed in the system boot code area, SDI mode, and RAM area, except for full chip erasure. User selectable word area can be erased or programmed. If you try to remove read protection (program user words), the chip will automatically erase the entire user area.

Note: The internal RC oscillator (HSI) must be turned on when performing flash program/erase operations.

14.3 Register Description

Table 14-2 FLASH registers

Name	Access address	Description	Reset value
R32_FLASH_ACTLR	0x40022000	Control register	0x00000000
R32_FLASH_KEYR	0x40022004	FPEC key register	X
R32_FLASH_OBKEYR	0x40022008	OBKEY register	X
R32_FLASH_STATR	0x4002200C	Status register	0x00008000
R32_FLASH_CTLR	0x40022010	Configuration register	0x00008080
R32_FLASH_ADDR	0x40022014	Address register	X
R32_FLASH_OBR	0x4002201C	Option byte register	0x0XXXXXXX
R32_FLASH_WPR	0x40022020	Write protection register	0xFFFFFFFF
R32_FLASH_MODEKEYR	0x40022024	Extension key register	X
R32_FLASH_BOOT_MODEKEYR	0x40022028	Unlock BOOT key register	X

14.3.1 Control Register (FLASH_ACTLR)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														LATENCY	

Bit	Name	Access	Description	Reset value
[31:2]	Reserved	RO	Reserved	0
[1:0]	LATENCY[1:0]	RW	FLASH wait state number. 00: 0 wait (recommended 0=<SYSCLK=<24MHz); 01: 1 wait (24<SYSCLK=<48MHz recommended); Other: invalid.	0

14.3.2 FPEC Key Register (FLASH_KEYR)

Offset address: 0x04

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEYR[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEYR[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	KEYR[31:0]	WO	FPEC key, the unlocking keys used to enter FPEC include: RDPRT key = 0x000000A5; KEY1 = 0x45670123; KEY2 = 0xCDEF89AB.	X

14.3.3 OBKEY Register (FLASH_OBKEYR)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

OBKEYR[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OBKEYR[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	OBKEYR[31:0]	WO	Option byte key, used to input select word key to release OBWRE.	X

14.3.4 Status Register (FLASH_STATR)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOCK	MODE	Reserved								EOP	WRPRTERR	Reserved			BSY

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	0
15	LOCK	RW	BOOT Lock 1: Locked, write operation cannot be executed to the FLASH_STATA[14] field. 0: Unlocked, write operation can be executed to FLASH_STATR[14] field. Note: Write 1 is set, write 0 is invalid.	1
14	MODE	RW	Can control switching between user area and BOOT area. 1: You can switch to the BOOT area after software reset; 0: You can switch to user area after software reset.	0
[13:6]	Reserved	RO	Reserved	0
5	EOP	RW1	Indicates the end of the operation and clears it by writing 1. Set by hardware on each successful erase or program.	0
4	WRPRTERR	RW1	Indicates a write protection error and is cleared by writing 1. This bit is set by hardware if a write-protected address is programmed.	0
[3:1]	Reserved	RO	Reserved	0
0	BUSY	RO	Indicates busy status. 1: Indicates that the flash memory operation is in progress; 0: The operation is completed.	0

Note: When performing programming operations, you need to ensure that the STRT bit of the FLASH_CTLR register is 0.

14.3.5 Configuration Register (FLASH_CTLR)

Offset address: 0x10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved												BUFRST	BUFLOAD	FTER	FTPG

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLOCK	Reserved		EOPIE	Reserved	ERRIE	OBWRE	Reserved	LOCK	STRT	OBER	OBPG	Reserved	MER	PER	PG

Bit	Name	Access	Description	Reset value
[31:20]	Reserved	RO	Reserved	0
19	BUFRST	RW	BUF reset operation.	0
18	BUFLOAD	RW	Cache data into BUF.	0
17	FTER	RW	Perform a fast page (64Byte) erase operation.	0
16	FTPG	RW	Perform fast page programming operations.	0
15	FLOCK	RW1	Fast programming lock. Only '1' can be written. When this bit is '1', it indicates that the fast program/erase mode is not available. After detecting the correct unlock sequence, the hardware clears this bit to '0'. The software sets it to 1 and re-locks it.	1
[14:13]	Reserved	RO	Reserved	0
12	EOPIE	RW	Operation completion interrupt control (EOP set in FLASH_STATR register). 1: Enable interrupts; 0: Disable interrupts.	0
11	Reserved	RO	Reserved	0
10	ERRIE	RW	Error status interrupt control (PGERR/WRPRTERR set in FLASH_STATR register). 1: Enable interrupts; 0: Disable interrupts.	0
9	OBWRE	RW0	User option byte lock and the software clears it to 0. 1: Indicates that the user-selected word can be programmed. It needs to be set by hardware after writing the correct sequence in the FLASH_OBKEYR register; 0: Re-lock the user-selected word after software clears it.	0
8	Reserved	RO	Reserved	0
7	LOCK	RW1	Lock. Only '1' can be written. When this bit is '1', it means that FPEC and FLASH_CTLR are locked and cannot be written. After detecting the correct unlock sequence, the hardware clears this bit to '0'. After an unsuccessful unlock operation, this bit will not change again until the next system reset.	1
6	STRT	RW1	Start. Set to 1 to start an erasure action, and the hardware will automatically clear it to 0 (BSY changes to '0').	0
5	OBER	RW	Perform user option byte erasure.	0
4	OBPG	RW	Perform user option byte programming.	0
3	Reserved	RO	Reserved	0
2	MER	RW	Perform a full wipe operation (erase the entire	0

			user area).	
1	PER	RW	Perform sector erase (1K).	0
0	PG	RW	Perform standard programming operations.	0

14.3.6 Address Register (FLASH_ADDR)

Offset address: 0x14

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FAR[31:16]															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FAR[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	FAR[31:0]	WO	The flash memory address is the programming address when programming, and the starting address of erasing when erasing. When the BSY bit in the FLASH_STATR register is '1', this register cannot be written.	0

14.3.7 Option Byte Register (FLASH_OBR)

Offset address: 0x1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved						DATA1								DATA0	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA0						FIX_11		START_MODE	RST_MODE		STANDY_RST	Reserved		RDPRT	OBERR

Bit	Name		Access	Description	Reset value
[31:26]	Reserved		RO	Reserved	0
[25:18]	DATA1[7:0]			Data byte 1	X
[17:10]	DATA0[7:0]			Data byte 0	X
[9:8]	FIX_11			Fixed to 11	11b
7	USER	START_MODE	RO	1: Start from the BOOT area. 0: Start from the user area. User applications can also be placed in the BOOT area.	1
[6:5]	USER	RST_MODE	RO	Configuration word reset delay time.	X
4	USER	STANDY_RST	RO	System reset control in standby mode.	X
3	USER	Reserved	RO	Reserved	X
2	USER	Reserved	RO	Reserved	X
1	RDPRT		RO	Read protection status. 1: Indicates that the current read protection of the flash memory is valid.	1
0	OBERR		RO	*Option byte error.* 1: Indicates that the selected word does not match its complement.	0

Note: USER and RDPRT are loaded from the user option byte area after system reset.

14.3.8 Write Protection Register (FLASH_WPR)

Offset address: 0x20

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WPR[15:0]															

Bit	Name	Access	Description	Reset value
[31:16]	Reserved	RO	Reserved	X
[15:0]	WPR[15:0]	RO	Flash write protection status. 1: Write protection is invalid; 0: Write protection is valid. Each bit represents 1K bytes (16 pages) of storage write protection status.	X

Note: WPR is loaded from the user option byte area after system reset.

14.3.9 Extension Key Register (FLASH_MODEKEYR)

Offset address: 0x24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODEKEYR[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODEKEYR[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	MODEKEYR[31:0]	WO	Enter the following sequence to unlock quick program/erase mode. KEY1 = 0x45670123; KEY2 = 0xCDEF89AB.	X

14.3.10 BOOT Key Register (BOOT_MODEKEYP)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODEKEYR[31:16]															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODEKEYR[15:0]															

Bit	Name	Access	Description	Reset value
[31:0]	MODEKEYR[31:0]	WO	Enter the following sequence to unlock BOOT area KEY1 = 0x45670123; KEY2 = 0xCDEF89AB.	X

14.4 Flash Operation Procedure

14.4.1 Read Operation

Direct addressing is performed in the general address space, and any read operation of 8/16/32-bit data can access

the contents of the flash memory module and obtain the corresponding data.

14.4.2 Unlock Flash Memory

After the system reset, the flash memory controller (FPEC) and FLASH_CTLR register will be locked and cannot be accessed. The flash memory controller module can be unlocked by writing the sequence to the FLASH_KEYR register.

Unlock sequence:

1） Write KEY1 = 0x45670123 to the FLASH_KEYR register (must operate KEY1 first);

2） Write KEY2 = 0xCDEF89AB to the FLASH_KEYR register (must operate KEY2 secondly).

The above operations must be performed sequentially and continuously. Otherwise, it is an error operation, which will lock the FPEC module and FLASH_CTLR register and generate a bus error until the next system reset.

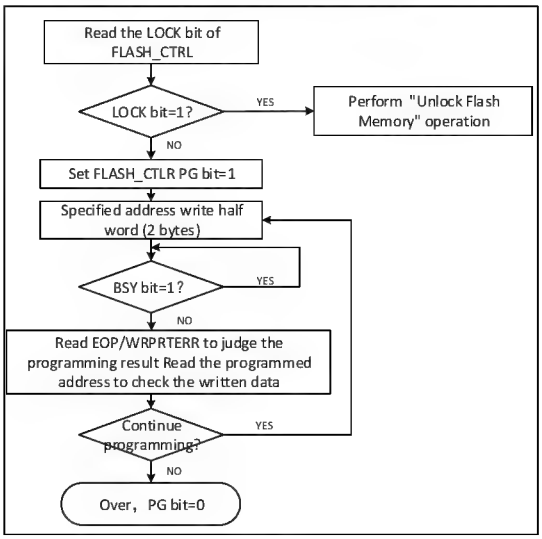
The flash memory controller (FPEC) and the FLASH_CTLR register can be locked again by setting the "LOCK" bit in the FLASH_CTLR register to 1.

14.4.3 Main Memory Standard Programming

Standard programming can write 2 bytes at a time. When the PG bit of the FLASH_CTLR register is '1', each time a halfword (2 bytes) is written to the flash memory address, a programming will be initiated. If any non-halfword data is written, FPEC will generate a bus error. During the programming process, the BSY bit is '1'. When programming is completed, the BSY bit is '0' and the EOP bit is '1'.

Note: When the BSY bit is '1', writing to any register is prohibited.

Figure 14-1 FLASH programming

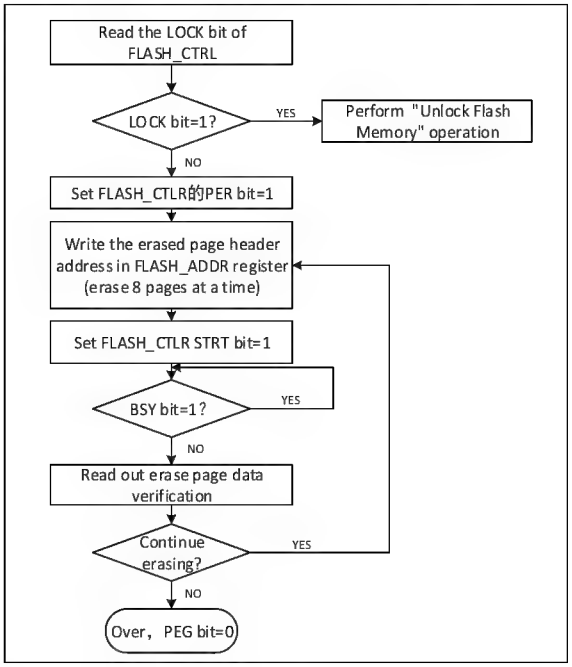


1) Check the FLASH_CTLR register LOCK. If it is 1, you need to perform the "unlock flash memory" operation.
2) Set the PG bit of the FLASH_CTLR register to '1' to enable the standard programming mode.
3) Write the halfword to be programmed to the specified flash address (even address).
4) Wait for the BSY bit to become '0' or the EOP bit of the FLASH_STATR register to be '1' to indicate the end of programming, and clear the EOP bit to 0.
5) Query the FLASH_STATR register to see if there is an error, or read the programming address data for verification.
6) To continue programming, you can repeat steps 3-5 and end programming by clearing the PG bit to 0.

14.4.4 Main Memory Standard Erase

Flash memory can be erased in standard pages (1K bytes) or in its entirety.

Figure 14-2 FLASH page erase



1) Check the LOCK bit of FLASH_CTLR register, if it is 1, you need to perform the "Unlock Flash" operation.

2) Set the PER bit of FLASH_CTLR register to '1' to enable the standard page erase mode.

3) Write the page header address to the FLASH_ADDR register to select the page to be erased.

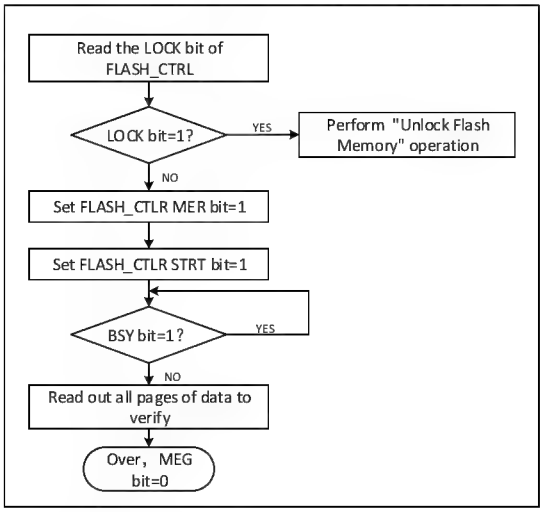
4) Set the STAT bit of FLASH_CTLR register to '1' to initiate an erase action.

5) Wait for the BYS bit to become '0' or the EOP bit of FLASH_STATR register to become '1' to indicate the end of erasure, and clear the EOP bit to 0.

6) Read the data of the erased page for verification.

7) Continue the standard page erase can repeat steps 3-5, end the erase to clear the PEG bit to 0.

Figure 14-3 FLASH whole chip erase



1) Check the LOCK bit of FLASH_CTLR register, if it is 1, you need to perform the "unlock flash memory" operation.

2) Set the MER bit of FLASH_CTLR register to '1' to enable the whole chip erase mode.

3) Set the STAT bit of FLASH_CTLR register to '1' to start the erase operation.
4) Wait for the BYS bit to become '0' or the EOP bit of FLASH_STATR register to be '1' to indicate the end of erasure, and clear the EOP bit to 0.
5) Read the data of the erased page for verification.
6) Clear the MER bit to 0.

14.4.5 Fast Programming Mode Unlock

Fast programming mode operation can be unlocked by writing a sequence to the FLASH_MODEKEYR register. After unlocking, the FLOCK bit in the FLASH_CTLR register will clear to 0, indicating that fast erase and programming operations can be performed. The FLASH_CTLR register can be locked again by software setting the "FLOCK" bit to 1.

Unlock sequence:

1) Write KEY1 = 0x45670123 to FLASH_MODEKEYR register;
2) Write KEY2 = 0xCDEF89AB to FLASH_MODEKEYR register.

The above operations must be performed in sequence and continuously, otherwise the system will be locked due to incorrect operations and cannot be unlocked again until the next system reset.

Note: Fast programming operation requires unlocking both "LOCK" and "FLOCK" locks.

14.4.6 Main Memory Fast Programming

Fast programming programs by page (64 bytes).

1) Check the LOCK bit of the FLASH_CTLR register. If it is 1, you need to perform a "release flash lock" operation.
2) Check the BSY bit of the FLASH_STATR register to confirm that there are no other programming operations in progress.
3) Check the FLOCK bit of the FLASH_CTLR register. If it is 1, you need to perform a "fast programming mode unlock" operation.
4) Set the FTPG bit of the FLASH_CTLR register to enable the fast page programming mode function.
5) Set the BUFRST bit of the FLASH_CTLR register to perform a clear internal 64 byte buffer operation.
6) Wait for the BYS bit to become '0' or the EOP bit of the FLASH_STATR register to be '1' to indicate the end of clearing, clear the EOP bit to 0.
7) Start writing 4 bytes of data to the specified address (4 bytes/operation), and then Set the BUFLOAD bit of the FLASH_CTLR register to perform loading into the buffer area.
8) Wait for the BYS bit to become '0' or the EOP bit of FLASH_STATR register to become '1' to indicate the end of clearing, and clear the EOP bit to 0
9) Repeat steps 7-8 a total of 16 times to load all 64-byte data into the cache area (the addresses of the 16 rounds of operations must be consecutive).
10) Write the first address of the fast page programming to the FLASH_ADDR register.
11) Set the STAT bit of the FLASH_CTLR register to '1' to start a fast page programming action.
12) Wait for the BSY bit to become '0' or the EOP bit of FLASH_STATR register to become '1' to indicate the completion of one fast page programming and clear the EOP bit to 0.
13) Query the FLASH_STATR register to see if there is an error, or read the programming address data for verification.
14) To continue fast page programming, you can repeat steps 5-13 and end programming by clearing the FTPG bit to 0.

14.4.7 Main Memory Fast Erase

Fast erase erases by page (64 bytes).

1) Check the LOCK bit of the FLASH_CTLR register. If it is 1, you need to perform a "release flash lock" operation.
2) Check the FLOCK bit of the FLASH_CTLR register. If it is 1, you need to perform a "fast programming mode unlock" operation.
3) Check the BSY bit of the FLASH_STATR register to confirm that there are no other programming operations in progress.
4) Set the FTER bit of the FLASH_CTLR register to '1' to enable the fast page erase (64-byte) mode function.
5) Write the first address of the flash erase page to the FLASH_ADDR register.
6) Set the STAT bit of the FLASH_CTLR register to '1' to start a fast page erase (64 bytes) action.
7) Wait for the BSY bit to become '0' or the EOP bit of FLASH_STATR register to become '1' to indicate the end of erase, and clear the EOP bit to 0.
8) Query the FLASH_STATR register to see if there is an error, or read the erase page address data to verify.
9) To continue fast page erase, you can repeat steps 5-8 and end the erase by clearing the FTER bit to 0.

14.5 User Option Bytes

The user option byte is solidified in FLASH and will be reloaded into the corresponding register after the system is reset. The user can erase and program at will. The user selection word information block has a total of 8 bytes (4 bytes for write protection, 1 byte for read protection, 1 byte for configuration options, and 2 bytes for user data storage). Each bit has its complement bit is used for verification during the loading process. The selection word information structure and meaning are described below.

Table 14-3 32-bit option byte format division

[31:24]	[23:16]	[15:8]	[7:0]
Inverse code of option bytes 1	Option bytes 1	Inverse code of option bytes 0	Option bytes 0

Table 14-4 User option bytes information structure

Address \ Bit	[31:24]	[23:16]	[15:8]	[7:0]
0x1FFFF800	nUSER	USER	nRDPR	RDPR
0x1FFFF804	nData1	Data1	nData0	Data0
0x1FFFF808	nWRPR1	WRPR1	nWRPR0	WRPR0
0x1FFFF80C	Reserved	Reserved	Reserved	Reserved

Name/Byte			Description	Reset value
RDPR			Read protection control. It configures whether the code in the flash memory can be read. 0xA5: If this byte is 0xA5 (nRDP must be 0x5A), it means that the current code is in a non-read protected state and can be read; Other values: Code read protection status, unreadable; pages 0 to 31 pages (4K) will be automatically write-protected and not controlled by WRPR0.	0x01
USER	[7:6]	Reserved	Reserved (must be 1)	11b
	5	START_M	Power-on startup method	1

		ODE	1: Start from the BOOT area 0: Start from the user area, the BOOT area can also place user applications.	
	[4:3]	RST_MODE [1:0]	PA6 is multiplexed as an external reset pin NRST. 00: Enable the multiplexing function, and the power-on reset will ignore the pin status within 128μs; 01: Enable the multiplexing function, and the power-on reset will ignore the pin status within 1ms; 10: Enable the multiplexing function, and the power-on reset will ignore the pin status within 12ms; 11: Multiplexing function Closed, PA6 has IO function.	11b
	2	STANDYRST	System reset control in standby mode: 1: Disabled, the system does not reset when entering standby mode; 0: Enabled, system reset occurs when entering standby mode.	1
	1	Reserved	Reserved	1
	0	Reserved	Reserved	1
Data0–Data1			Stores 2 bytes of user data.	FFFFh
WRPR0 - WRPR3			Write protection control bit. Each bit is used to control the write protection status of 1 sector (1K bytes/sector) in main memory. 1: Enable write protection; 0: Enable write protection. 2 bytes are used to protect a total of 16K bytes of main memory. WRPO: Sectors 0-7 store write protection control; WRP1: Sectors 8-15 store write protection control; WRP2: Reserved; WRP3: Reserved.	FFFFh

14.5.1 User Option Bytes Unlock

User option byte operation can be unlocked by writing a sequence to the FLASH_OBKEYR register. After unlocking, the OBWRE bit of the FLASH_CTLR register will be set to 1, indicating that erasing and programming of user-selected words can be performed. Lock again by clearing the "OBWRE" bit of the FLASH_CTLR register to 0 by software.

Unlock sequence:

1) Write KEY1 = 0x45670123 to the FLASH_OBKEYR register;
2) Write KEY2 = 0xCDEF89AB to the FLASH_OBKEYR register.

Note: The user needs to unlock the 2 layers: "LOCK" and "OBWRE" for word selection.

14.5.2 User Option Bytes Programming

Only supports standard programming, writing half word (2 bytes) at a time. In the actual process, when programming the user option byte, FPEC only uses the low byte in the half word, and automatically calculates the high byte (the high byte is the complement of the low byte), and then starts the programming operation, which will It is guaranteed that the byte in the user-selected word and its complement are always correct.

1) Check the LOCK bit of the FLASH_CTLR register. If it is 1, you need to perform the "unlock flash lock" operation.
2) Check the BSY bit of the FLASH_STATR register to confirm that there are no other programming operations in progress.

3) Check the OBWRE bit of the FLASH_CTLR register. If it is 0, the "user selected word unlock" operation needs to be performed.
4) Set the OBPG bit of the FLASH_CTLR register to '1'.
5) Write the halfword (2 bytes) to be programmed to the specified address.
6) Wait for the BYS bit to become '0' or the EOP bit of the FLASH_STATR register to be '1' to indicate the end of programming, and clear the EOP bit to zero.
7) Read programming address data verification.
8) To continue programming, you can repeat steps 5-7 and end programming by clearing the OBPG bit to 0.

Note: When the "read protection" in the modified selection word changes to the "unprotected" state, a full chip erase operation of the main storage area will be automatically performed. If you modify the selection other than "read protection", the entire chip erase operation will not occur.

14.5.3 User Option Bytes Erase

Directly erase the entire 64-byte user option byte area.

1) Check the LOCK bit of the FLASH_CTLR register. If it is 1, you need to perform a "release flash lock" operation.
2) Check the BSY bit of the FLASH_STATR register to confirm that no programming operation is in progress.
3) Check the OBWRE bit of the FLASH_CTLR register. If it is 0, the "user option byte unlock" operation needs to be performed.
4) Set the OBER bit of the FLASH_CTLR register to '1' and later set the STAT bit of the FLASH_CTLR register to '1' to enable user option byte erase.
5) Wait for the BYS bit to become '0' or the EOP bit of the FLASH_STATR register to be '1' to indicate the end of erasure, and clear the EOP bit to zero.
6) Read erase address data verification.
7) End clears the OBER bit to 0.

14.5.4 Release Read Protection

Whether the flash memory is read-protected or not is determined by the user option byte. Read the FLASH_OBR register. When the RDPRT bit is '1', it means that the current flash memory is in the read protection state, and the flash memory operation is subject to a series of security protections in the read protection state. The process of lifting read protection is as follows:

1) Erase the entire user option byte area. At this time, the read protection field RDPR is still valid.
2) The user selects word programming and writes the correct RDPR code 0xA5 to release the read protection of the flash memory. (This step will first cause the system to automatically perform a full chip erase operation on the flash memory)
3) Perform a power-on reset to reload the selection byte (including the new RDPR code), and the read protection is released at this time.

14.5.5 Release Write Protection

Whether the flash memory is write-protected or not is determined by the user option byte. Read the FLASH_WPR register. Each bit represents 4K bytes of flash memory space. When the bit is '1', it indicates non-write protection status, and when it is '0', it indicates write protection. The process of removing write protection is as follows:

1) Erases the entire user option byte area.
2) Write the correct RDPR code 0xA5 to allow read access.

3) Perform a system reset, reload the selection bytes (including the new WRPR[3:0] bytes), and the write protection is released.

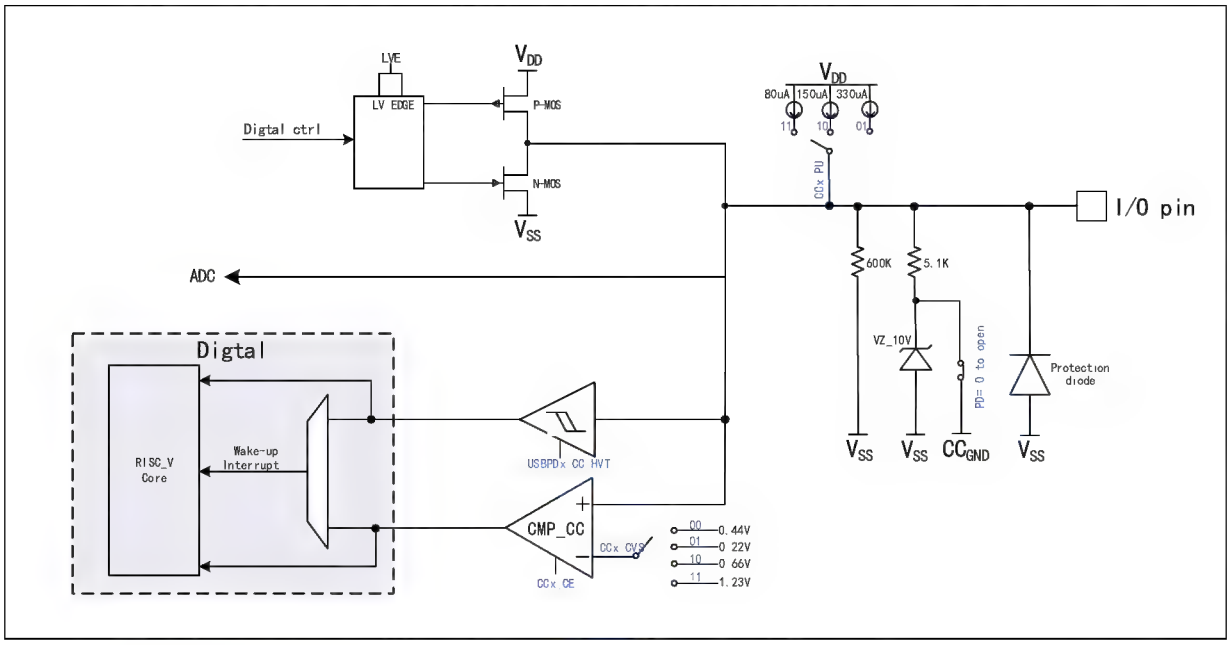
Chapter 15 USB PD Controller (USBPD)

15.1 Introduction to USB PD Controller

The chip has a built-in USB Power Delivery controller and PD transceiver PHY, and provides 3 CC pins, of which PB0/CC1R and PB9/CC3R pins have built-in type-C specification-defined controllable Rd pull-down resistors, and PB1/CC2 pins do not provide Rd by default, and customization can be supported.
Support USB Type-C master-slave detection, automatic BMC codec and CRC, hardware edge control, USB PD2.0 and PD3.0 power transfer control, fast charging, UFP/PD Sink and DFP/PD Source application, DRP application and dynamic switching. Built-in USB Type-C interface, supports master-slave detection, DRP, Sink/Consumer and Source/Provider.

- Built-in USB PD transceiver PHY with integrated hardware edge-slope control;
- Built-in USB Power Delivery controller, automatic BMC codec, 4b5b codec and CRC;
- Support SOP, SOP', SOP" PD packets and USB PD reset signal frame hardware reset;
- Support maximum packet length of 510 bytes, DMA;
- Support USB PD 2.0 and 3.0 power transfer protocols, USB port supports BC and other charging protocols.
- Support comparator and multiple optional negative input reference voltages, programmable pull-up current

Figure 15-1 CC port structure diagram



Note: CCGND is the internal GND of CCx. If CCx has suffix R, that is, CCxR, its CCGND has been shorted to VSS inside the chip.

15.2 Register Description

Table 15-1 USBPD-related registers

Name	Access address	Description	Reset value
R32_USBPD_CONFIG	0x40024000	PD configuration register	0x00000002
R16_CONFIG	0x40024000	PD interrupt enable register	0x0002

R16_BMC_CLK_CNT	0x40024002	BMC sample clock counter	0x0000
R32_USBPD_CONTROL	0x40024004	PD control register	0x00000000
R16_CONTROL	0x40024004	PD transceiver control register	0x0000
R8_CONTROL	0x40024004	PD transceiver enable register	0x00
R8_TX_SEL	0x40024005	PD transmit SOP selection register	0x00
R16_BMC_TX_SZ	0x40024006	PD transmit length register	0x0000
R32_USBPD_STATUS	0x40024008	PD status register	0x000000XX
R16_STATUS	0x40024008	PD interrupt and data register	0x00XX
R8_DATA_BUF	0x40024008	DMA cache data register	0xXX
R8_STATUS	0x40024009	PD interrupt flag register	0x00
R16_BMC_BYTE_CNT	0x4002400A	Byte counter	0x0000
R32_USBPD_PORT	0x4002400C	Port control register	0x03030003
R8_PORT_CC1	0x4002400C	CC1 port control register	0x03
R8_PORT_CC2	0x4002400E	CC2 port control register	0x03
R8_PORT_CC3	0x4002400F	CC3 port control register	0x03
R32_USBPD_DMA	0x40024010	DMA cache address register	0x0000XXXX
R16_DMA	0x40024010	PD buffer start address register	0xXXXX

15.2.1 Configuration Register (R32_USBPD_CONFIG)

Offset address: 0x00

Bit	Name
[31:16]	R16_BMC_CLK_CNT
[15:0]	R16_CONFIG

15.2.2 PD Interrupt Enable Register (R16_CONFIG)

Offset address: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE_TX_END	IE_RX_RESET	IE_RX_ACT	IE_RX_BYTE	IE_RX_BYTE	IE_PD_IO	Reserved			WAKE_POLAR	PD_RST_EN	PD_DMA_EN	CC_SEL [1:0]		PD_ALL_CLR	CC_FILTER_EN

Bit	Name	Access	Description	Reset value
15	IE_TX_END	RW	Transmit end interrupt enable.	0
14	IE_RX_RESET	RW	Receive reset interrupt enable.	0
13	IE_RX_ACT	RW	Receive complete interrupt enable.	0
12	IE_RX_BYTE	RW	Receive byte interrupt enable.	0
11	IE_RX_BIT	RW	Receive bit interrupt enable.	0
10	IE_PD_IO	RW	PD IO interrupt enable.	0
[9:7]	Reserved	RO	Reserved.	0
6	WAKE_POLAR	RW	PD port wake-up level: 0: Active low; 1: Active high.	0
5	PD_RST_EN	RW	PD mode reset command enable: 0: Invalid; 1: Reset.	0
4	PD_DMA_EN	RW	Enable DMA for USBPD, this bit must be set to 1 in normal transmission mode: 1: Enable DMA function and DMA interrupt; 0: Disable DMA.	0
[3:2]	CC_SEL[1:0]	RW	Select the current PD communication port: 00: Use CC1 port for communication; 01: Use CC2 port for communication;	0

			1x: Use CC3 port for communication.	
1	PD_ALL_CLR	RW	PD mode clears all interrupt flag bits: 0: invalid; 1: use CC2 port 0: Invalid; 1: Clear the interrupt flag bits.	1
0	CC_FILTER_EN	RW	Control the input filter enable on the PD pin: 0: Turn off filtering; 1: Turn on filtering.	X

15.2.3 BMC Sample Clock Counter (R16_BMC_CLK_CNT)

Offset address: 0x02

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							BMC_CLK_CNT								

Bit	Name	Access	Description	Reset value
[15:9]	Reserved	RO	Reserved	0
[8:0]	BMC_CLK_CNT	RW	The BMC transmits or receives a sample clock counter.	0

15.2.4 Control Register (R32_USBPD_CONTROL)

Offset address: 0x04

Bit	Name
[31:16]	R16_BMC_TX_SZ
[15:0]	R16_CONTROL

15.2.5 PD Transceiver Control Register (R16_CONTROL)

Offset address: 0x04

Bit	Name
[15:8]	R8_TX_SEL
[7:0]	R8_CONTROL

15.2.6 PD Transceiver Enable Register (R8_CONTROL)

Offset address: 0x04

7	6	5	4	3	2	1	0
BMC_BYTE_HI	TX_BIT_BACK	DATA_FLAG	RX_STATE			BMC_START	PD_TX_EN

Bit	Name	Access	Description	Reset value
7	BMC_BYTE_HI	RO	Indicates the current half-byte status of the PD data being sent and received: 0: The lower 4 bits are being processed; 1: Indicates that the high 4 bits are being	0
6	TX_BIT_BACK	RO	Indicates the current bit status of the BMC when sending the code: 0: Idle; 1: Indicate that a BMC byte is being transmitted.	0
5	DATA_FLAG	RO	Cache data valid flag bit.	0
[4:2]	RX_STATE	RO	PD receive status identification 000: Receive initial status 001: Start receiving SOP 010: Receive reset 011: Receive SOP	0

			100: Receive end 101: Receive unused 110: Receive EOP 111: Receive byte.	
1	BMC_START	RW	BMC transmit start signal.	0
0	PD_TX_EN	RW	USBPD receive and transmit mode and transmit enable: 0: PD receive enable; 1: PD transmit enable.	0

15.2.7 PD Transmit SOP Selection Register (R8_TX_SEL)

Offset address: 0x05

7	6	5	4	3	2	1	0
TX_SEL4		TX_SEL3		TX_SEL2		Reserved	TX_SEL1

Bit	Name	Access	Description	Reset value
[7:6]	TX_SEL4	RW	K-CODE4 type selection in PD transmit mode: 00: SYNC2; 01: SYNC3; 1x: RST2.	0
[5:4]	TX_SEL3	RW	K-CODE3 type selection in PD transmit mode: 00: SYNC1; 01: SYNC3; 1x: RST1.	0
[3:2]	TX_SEL2	RW	K-CODE2 type selection in PD transmit mode: 00: SYNC1; 01: SYNC3; 1x: RST1.	0
1	Reserved	RO	Reserved	0
0	TX_SEL1	RW	K-CODE1 type selection in PD transmit mode: 0: SYNC1; 1: RST1.	0

15.2.8 PD Transmit Length Register (R16_BMC_TX_SZ)

Offset address: 0x06

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							BMC_TX_SZ								

Bit	Name	Access	Description	Reset value
[15:9]	Reserved	RO	Reserved	0
[8:0]	BMC_TX_SZ	RW	The total length sent in PD mode.	0

15.2.9 Status Register (R32_USBPD_STATUS)

Offset address: 0x08

Bit	Name
[31:16]	R16_BMC_BYTE_CNT
[15:0]	R16_STATUS

15.2.10 PD Interrupt and Data Register (R16_STATUS)

Offset address: 0x08

Bit	Name

[15:8]	R8_STATUS
[7:0]	R8_DATA_BUF

15.2.11 DMA Buffer Data Register (R8_DATA_BUF)

Offset address: 0x08

7	6	5	4	3	2	1	0
DATA_BUF							

Bit	Name	Access	Description	Reset value
[7:0]	DATA_BUF	RO	DMA buffer data.	X

15.2.12 PD Interrupt Flag Register (R8_STATUS)

Offset address: 0x09

7	6	5	4	3	2	1	0
IF_TX_END	IF_RX_RESET	IF_RX_ACT	IF_RX_BYTE	IF_RX_BIT	BUF_ERR	BMC_AUX	

Bit	Name	Access	Description	Reset value
7	IF_TX_END	RW1	Transmit complete interrupt flag, write 1 to clear 0, write 0 is invalid.	0
6	IF_RX_RESET	RW1	Receive reset interrupt flag, write 1 to clear 0, write 0 is invalid.	0
5	IF_RX_ACT	RW1	Receive complete interrupt flag, write 1 to clear 0, write 0 is invalid.	0
4	IF_RX_BYTE	RW1	Receive byte or SOP interrupt flag, write 1 to clear 0, write 0 is invalid.	0
3	IF_RX_BIT	RW1	Receive bit or 5bit interrupt flag, write 1 to clear 0, write 0 is invalid.	0
2	BUF_ERR	RW1	BUFFER or DMA error interrupt flag, write 1 to clear 0, write 0 is invalid.	0
[1:0]	BMC_AUX	RO	Indicates the current PD status: During PD reception, the status is as follows: 00: reception idle or no valid packet received; 01: SOP received i.e. SOP0; 10: SOP' received i.e. SOP1; 11: SOP' received i.e. SOP2. When PD is sending, the status is as follows: 00: CRC32[7:0] is being transmitted; 01: CRC32[15:8] is being transmitted; 10: CRC32[23:16] is being transmitted; 11: CRC32[31:24] is being transmitted.	00b

15.2.13 Byte Counter (R16_BMC_BYTE_CNT)

Offset address: 0x0A

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							BMC_BYTE_CNT								

Bit	Name	Access	Description	Reset value
[15:9]	Reserved	RO	Reserved.	0
[8:0]	BMC_BYTE_CNT	RO	Byte counter.	0

15.2.14 Port Control Register (R32_USBPD_PORT)

Offset address: 0x0C

Bit	Name
[31:24]	R8_PORT_CC3
[23:16]	R8_PORT_CC2
[15:8]	Reserved
[7:0]	R8_PORT_CC1

15.2.15 CC1 Port Control Register (R16_PORT_CC1)

Offset address: 0x0C

7	6	5	4	3	2	1	0
CC1_CE	CC1_CVS		CC1_LVE	CC1_PU		CC1_PD	CC1_CMPO

Bit	Name	Access	Description	Reset value
7	CC1_CE	RW	Enable the CC1 port voltage CMP: 0: Disable the CMP, default CC1_CMPO output 1; 1: Enable the CMP, select the reference voltage by CC1_CVS.	0
[6:5]	CC1_CVS	RW	Select the reference voltage of CC1 port voltage CMP: 00: 0.55V; 01: 0.22V; 10: 0.66V; 11: 1.23V.	00b
4	CC1_LVE	RW	CC1 port output low voltage enable: 0: Normal V_{DD} voltage drive output; 1: PD low voltage drive output.	0
[3:2]	CC1_PU	RW	CC1 port pull-up current selection (need to enable CC1_REF bit) 00: Disable pull-up current; 01: Rated 330μA; 10: Rated 180μA; 11: Rated 80μA.	00b
1	CC1_PD	RW	CC1 port pull-down resistor Rd enable: 0: Disable the pull-down resistor (Note: There is still a weak pull-down of about 600kΩ after turning off) 1: Enable the Rd pull-down resistor, about 5.1KΩ. Note: Some chips do not have built-in Rd, please refer to the selection table in CH641DS0.PDF	1
0	CC1_CMPO	RO	The result output bits of the CC1 voltage CMP: 0: The current port voltage is less than the reference voltage; 1: The current port voltage is greater than the reference voltage.	1

15.2.16 CC2 Port Control Register (R16_PORT_CC2)

Offset address: 0x0E

7	6	5	4	3	2	1	0

CC2_CE	CC2_CVS	CC2_LVE	CC2_PU	CC2_PD	CC2_CMPO

Bit	Name	Access	Description	Reset value
7	CC2_CE	RW	Enable the CC2 port voltage CMP: 0: Disable the CMP, default CC2_CMPO output 1; 1: Enable the CMP and select the reference voltage by CC2_CVS.	0
[6:5]	CC2_CVS	RW	Select the reference voltage of the CC2 port voltage CMP: 00: 0.55V; 01: 0.22V; 10: 0.66V; 11: 1.23V.	00b
4	CC2_LVE	RW	CC2 port output low voltage enable: 0: Normal V_{DD} voltage drive output; 1: PD low voltage drive output.	0
[3:2]	CC2_PU	RW	CC2 port pull-up current selection (need to enable CC2_REF bit) 00: Pull-up current disable; 01: 330μA; 10: 180μA; 11: 80μA.	00b
1	CC2_PD	RW	CC2 port pull-down resistor Rd is enabled: 0: Disable the pull-down resistor *(Note: There is still a weak pull-down of about 600kΩ after turning off)* 1: Enable the Rd pull-down resistor, about 5.1KΩ.	1
0	CC2_CMPO	RO	The result output bits of the CC2 voltage CMP: 0: The current port voltage is less than the reference voltage; 1: The current port voltage is greater than the reference voltage.	1

15.2.17 CC3 Port Control Register (R16_PORT_CC3)

Offset address: 0x0E

Bit	Name	Access	Description	Reset value
7	CC3_CE	RW	Enable the CC3 port voltage CMP: 0: Disable the CMP, default CC3_CMPO output 1; 1: Enable the CMP and select the reference voltage by CC3_CVS.	0
[6:5]	CC3_CVS	RW	Select the reference voltage of the CC3 port voltage CMP: 00: 0.55V; 01: 0.22V; 10: 0.66V; 11: 1.23V.	00b
4	CC3_LVE	RW	CC3 port output low voltage enable: 0: Normal V_{DD} voltage drive output; 1: PD low voltage drive output.	0
[3:2]	CC3_PU	RW	CC3 port pull-up current selection (need to enable CC2_REF bit) 00: Pull-up current disable; 01: 330μA; 10: 180μA;	00b

			11: 80μA.	
1	CC3_PD	RW	CC3 port pull-down resistor Rd is enabled: 0: Disable the pull-down resistor *(Note: There is still a weak pull-down of about 600kΩ after turning off)* 1: Enable the Rd pull-down resistor, about 5.1KΩ.	1
0	CC3_CMPO	RO	The result output bits of the CC3 voltage CMP: 0: The current port voltage is less than the reference voltage; 1: The current port voltage is greater than the reference voltage.	1

15.2.18 DMA Cache Address Register (R32_USBPD_DMA)

Offset address: 0x10

Bit	Name
[31:16]	Reserved
[15:0]	R16_DMA

15.2.19 PD Buffer Start Address Register (R16_DMA)

Offset address: 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USBPD_DMA_ADDR															

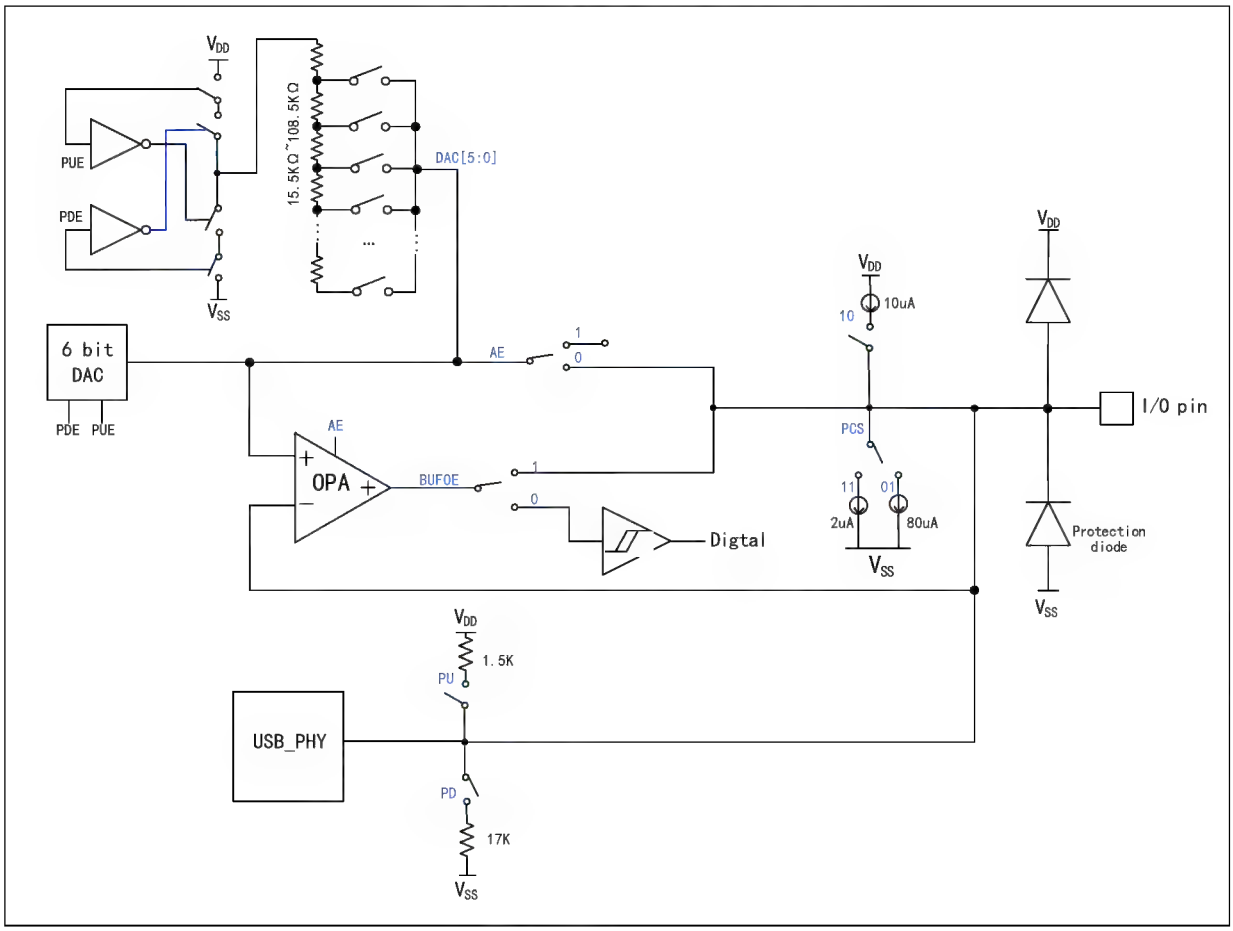
Bit	Name	Access	Description	Reset value
[15:0]	USBPD_DMA_ADDR	RW	USBPD_DMA cache address. The lower 16 bits are valid and the address must be 4-byte aligned.	X

Chapter 16 BC/Differential amplification/QII/Extended Configuration (EXTEN)

16.1 Extended Configuration

The system provides three EXTEN expansion configuration units (EXTEN_CTLR0 register, EXTEN_CTLR1 register, EXTEN_CTLR2 register). This unit uses the HB clock and only performs reset actions during system reset.
The EXTEN_CTLR0 register includes the LOCKUP function monitoring function: enabling the LKUPEN field will turn on the system's Lock-up situation monitoring. Once a Lock-up situation occurs, the system will perform a software reset and set the LKUPRST field to 1. After reading, you can write 1 to clear this logo.
The EXTEN_CTLR1 register includes the control bits for the BC interface UDM/UDP pins, the DAC and DAC_BUF output buffers/comparators, and the port pull-up/pull-down current.
The EXTEN_CTLR2 register includes control bits for differential input current sampling (ISP), AC small signal amplification decoder (QII), and auxiliary control bits for USBPD and CC pins.
Each pin of the BC interface contains a 6-bit DAC and an output buffer DAC_BUF. Among them, the DAC can not only output a programmable voltage, but also realize the pull-up or pull-down of the programmable resistor; DAC_BUF can not only buffer the DAC voltage and output it to the pin, but can also be used as an analog voltage comparator to compare the pin voltage with the DAC voltage.
Differential input current sampling will be an external resistor to capture the current obtained by the weak voltage signal through the closed-loop amplifier ISP_OP amplified results through the ADC_IN8 channel into the ADC, ADC sampling results can be used for upper limit comparisons, calculations and analysis.
AC small signal amplifier decoder includes the adjustable gain amplifier QII_OP in the front stage and the voltage comparator QII_CMP in the back stage, which can amplify the input AC small signal, shape it into a digital signal, filter it and decode it after processing, and the decoded result is sent to ADC for sampling through ADC_IN9 channel.

Figure 16-1 BC interface structure diagram



16.2 Function Description

As shown in the BC interface structure diagram in Figure 16-1, the following modes can be configured through the UDM/UDP corresponding control bits in the EXTEN_CTLR1 register:

DAC mode

1. When PDE=1, PUE=1, AE=0, DAC[5:0] input data, BUFOE=0, this is the DAC mode without buffer.
2. When PDE=1, PUE=1, AE=1, DAC[5:0] input data, BUFOE=1, this is DAC mode with buffer.

The DAC mode output voltage DAC_OUT is:

DAC_OUT=DAC[5:0]*VDD33/64

CMP mode

When PDE=1, PUE=1, AE=1, DAC[5:0] inputs data, and BUFOE=0, it is in CMP mode. The input of the positive terminal of the CMP is DAC_OUT, the input voltage of the negative terminal of the CMP is IO port, and the output signal of the CMP is sent to the digital terminal.

Pull-up/pull-down resistor mode

1. When PDE=0, PUE=1, AE=0, DAC[5:0] input data, BUFOE=0, this is the programmable pull-up resistor mode. Where, when DAC[5:0]=000001, the pull-up resistor is equal to 7*RDAC (about 108.5kΩ), and when DAC[5:0]=111111, the pull-up resistor is equal to 1366/1365*RDAC (about 15.5kΩ).
2. When PDE=1, PUE=0, AE=0, DAC[5:0] input data, BUFOE=0, this is programmable pull-down resistor mode.

In this case, when DAC[5:0]=000000, the pull-down resistor is equal to RDAC (about 15.5kΩ), and when DAC[5:0]=111111, the pull-down resistor is equal to 7*RDAC (about 108.5kΩ).

Note: See Table 16-1 for pull-up/down resistor resistance values.

Pull-up/pull-down current mode

1. When PCS = 10, this time for the weak pull-up current IPU2, about 10uA.
2. When PCS = 11, this time for the weak pull-down current IPD3, about 2uA.
3. When PCS = 01, this time for the weak pull-down current IPD1, about 80uA.

Table 16-1 Programmable pull-up/pull-down resistor value query table

DAC<5:0>	Pull-up resistor value (KΩ)	Pull-down resistor value (KΩ)	DAC<5:0>	Pull-up resistor value (KΩ)	Pull-down resistor value (KΩ)
000000	Invalid	$1*R_{DAC}$	100000	$2*R_{DAC}$	$2*R_{DAC}$
000001	$7*R_{DAC}$	$1366/1365*R_{DAC}$	100001	$14/9*R_{DAC}$	$683/341*R_{DAC}$
000010	$6*R_{DAC}$	$342/341*R_{DAC}$	100010	$3/2*R_{DAC}$	$171/85*R_{DAC}$
000011	$26/5*R_{DAC}$	$214/213*R_{DAC}$	100011	$13/9*R_{DAC}$	$107/53*R_{DAC}$
000100	$5*R_{DAC}$	$86/85*R_{DAC}$	100100	$10/7*R_{DAC}$	$43/21*R_{DAC}$
000101	$13/3*R_{DAC}$	$226/223*R_{DAC}$	100101	$26/19*R_{DAC}$	$113/55*R_{DAC}$
000110	$21/5*R_{DAC}$	$54/53*R_{DAC}$	100110	$42/31*R_{DAC}$	$27/13*R_{DAC}$
000111	$85/21*R_{DAC}$	$130/127*R_{DAC}$	100111	$170/127*R_{DAC}$	$65/31*R_{DAC}$
001000	$4*R_{DAC}$	$22/21*R_{DAC}$	101000	$4/3*R_{DAC}$	$11/5*R_{DAC}$
001001	$24/7*R_{DAC}$	$234/223*R_{DAC}$	101001	$24/19*R_{DAC}$	$117/53*R_{DAC}$
001010	$10/3*R_{DAC}$	$58/55*R_{DAC}$	101010	$5/4*R_{DAC}$	$29/13*R_{DAC}$
001011	$42/13*R_{DAC}$	$18/17*R_{DAC}$	101011	$21/17*R_{DAC}$	$9/4*R_{DAC}$
001100	$16/5*R_{DAC}$	$14/13*R_{DAC}$	101100	$16/13*R_{DAC}$	$7/3*R_{DAC}$
001101	$40/13*R_{DAC}$	$146/135*R_{DAC}$	101101	$40/33*R_{DAC}$	$73/31*R_{DAC}$
001110	$64/21*R_{DAC}$	$34/31*R_{DAC}$	101110	$64/33*R_{DAC}$	$17/7*R_{DAC}$
001111	$256/85*R_{DAC}$	$10/9*R_{DAC}$	101111	$256/213*R_{DAC}$	$5/2*R_{DAC}$
010000	$3*R_{DAC}$	$6/5*R_{DAC}$	110000	$6/5*R_{DAC}$	$3*R_{DAC}$
010001	$5/2*R_{DAC}$	$256/213*R_{DAC}$	110001	$10/9*R_{DAC}$	$256/85*R_{DAC}$
010010	$17/7*R_{DAC}$	$64/33*R_{DAC}$	110010	$34/31*R_{DAC}$	$64/21*R_{DAC}$
010011	$73/31*R_{DAC}$	$40/33*R_{DAC}$	110011	$146/135*R_{DAC}$	$40/13*R_{DAC}$
010100	$7/3*R_{DAC}$	$16/13*R_{DAC}$	110100	$14/13*R_{DAC}$	$16/5*R_{DAC}$
010101	$9/4*R_{DAC}$	$21/17*R_{DAC}$	110101	$18/17*R_{DAC}$	$42/13*R_{DAC}$
010110	$29/13*R_{DAC}$	$5/4*R_{DAC}$	110110	$58/55*R_{DAC}$	$10/3*R_{DAC}$
010111	$117/53*R_{DAC}$	$24/19*R_{DAC}$	110111	$234/223*R_{DAC}$	$24/7*R_{DAC}$
011000	$11/5*R_{DAC}$	$4/3*R_{DAC}$	111000	$22/21*R_{DAC}$	$4*R_{DAC}$
011001	$65/31*R_{DAC}$	$170/127*R_{DAC}$	111001	$130/127*R_{DAC}$	$85/21*R_{DAC}$
011010	$27/13*R_{DAC}$	$42/31*R_{DAC}$	111010	$54/53*R_{DAC}$	$21/5*R_{DAC}$
011011	$113/55*R_{DAC}$	$26/19*R_{DAC}$	111011	$226/223*R_{DAC}$	$13/3*R_{DAC}$
011100	$43/21*R_{DAC}$	$10/7*R_{DAC}$	111100	$86/85*R_{DAC}$	$5*R_{DAC}$
011101	$107/53*R_{DAC}$	$13/9*R_{DAC}$	111101	$214/213*R_{DAC}$	$26/5*R_{DAC}$
011110	$171/85*R_{DAC}$	$3/2*R_{DAC}$	111110	$342/341*R_{DAC}$	$6*R_{DAC}$
011111	$683/341*R_{DAC}$	$14/9*R_{DAC}$	111111	$1366/1365*R_{DAC}$	$7*R_{DAC}$

Note: R_{DAC} is the DAC input impedance.

16.3 Register Description

Table 16-2 EXTEN-related registers

Name	Access address	Description	Reset value
R32_EXTEN_CTLR0	0x40023800	Configure extended control register 0	0x00000040
R32_EXTEN_CTLR1	0x40023808	Configure extended control register 1	0xX010X010
R32_EXTEN_CTLR2	0x4002380C	Configure extended control register 2	0x00000000

16.3.1 Configure Extended Control Register 0 (EXTEN_CTLR0)

Offset address: 0x00

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								LKUPRST	LKUPEN	Reserved					

Bit	Name	Access	Description	Reset value
[31:8]	Reserved	RO	Reserved	0
7	LKUPRST	RW1	LOCKUP reset flag. 1: A LOCKUP occurs and causes a system reset. Write 1 to clear it; 0: Normal.	0
6	LKUPEN	RW	LOCKUP monitoring function enable bit: 1: Enabled, the system performs a reset when lock_up occurs, position LOCKUP_RSTF; 0: Disable.	1
[5:0]	Reserved	RO	Reserved	0

16.3.2 Configure Extended Control Register 1 (EXTEN_CTLR1)

Offset address: 0x08

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			UDM_AI	UDM_AE	UDM_DAC						UDM_PDE	UDM_PUE	UDM_PCS		UDM_BUFOE

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			UDP_AI	UDP_AE	UDP_DAC						UDP_PDE	UDP_PUE	UDP_PCS		UDP_BUFOE

Bit	Name	Access	Description	Reset value
[31:29]	Reserved	RO	Reserved	0
28	UDM_AI	RO	Output of the UDM pin comparator DAC_BUF.	x
27	UDM_AE	RW	UDM pin buffer/comparator DAC_BUF enable: 1: Output buffer/input comparator enabled; 0: Output buffer/input comparator disabled.	0
[26:21]	UDM_DAC	RW	UDM pin 6-bit DAC data or control data for programmable pull-up or pull-down resistors.	0

20	UDM_PDE	RW	UDM port DAC and programmable pull-up and pull-down mode selection: 00: disable DAC, disable programmable pull-up and pull-down resistors; PDE=0, PUE=1: programmable pull-up resistor mode; PDE=1, PUE=0: Programmable pull-down resistor mode; 11: DAC programmable voltage mode, the reference voltage for the comparator is either directly output to the pin or output to the pin through the buffer. If UDM_AE=0, it is directly output to the pin. If UDM_AE =1 and UDM_BUFOE=1, the buffer is output to the pin. If UDM_AE=1 and UDM_BUFOE=0, the pin voltage is compared with the DAC voltage, and the result is UDM_AI.	1
19	UDM_PUE	RW		0
[18:17]	UDM_PCS	RW	Select the built-in pull-up current or pull-down current of the UDM pin: 00: No pull-up, no pull-down. 01: Rated pull-down 2μA. 10: Rated pull-up 10μA. 11: Rated pull-down 80μA.	00b
16	UDM_BUFOE	RW	UDM pin DAC_BUF output enable: 1: UDM pin DAC_BUF output enable; 0: UDM pin DAC_BUF output disable.	0
[15:13]	Reserved	RO	Reserved	0
12	UDP_AI	RO	Output of the UDP pin comparator DAC_BUF.	x
11	UDP_AE	RW	UDP pin buffer/comparator DAC_BUF enable: 1: Output buffer/input comparator enabled; 0: Output buffer/input comparator disabled.	0
[10:5]	UDP_DAC	RW	UDP pin 6-bit DAC data or control data for programmable pull-up or pull-down resistors.	0
4	UDP_PDE	RW	UDP port DAC and programmable pull-up and pull-down mode selection: 00: disable DAC, disable programmable pull-up and pull-down resistors; PDE=0, PUE=1: programmable pull-up resistor mode; PDE=1, PUE=0: Programmable pull-down resistor mode; 11: DAC programmable voltage mode, the reference voltage for the comparator is either directly output to the pin or output to the pin via a buffer. If UDP_AE=0, it is directly output to the pin. If UDP_AE =1 and UDP_BUFOE=1, the buffer is output to the pin. If UDP_AE=1 and UDP_BUFOE=0, the pin voltage is compared with the DAC voltage, and the result is UDP_AI.	1
3	UDP_PUE	RW		0
[2:1]	UDP_PCS	RW	Select the built-in pull-up current or pull-down current of the UDP pin: 00: No pull-up, no pull-down. 01: Rated pull-down 2μA. 10: Rated pull-up 10μA. 11: Rated pull-down 80μA.	00b
0	UDP_BUFOE	RW	UDP pin DAC_BUF output enable: 1: UDP pin DAC_BUF output enable; 0: UDP pin DAC_BUF output disable.	0

16.2.3 Configure Extended Control Register 2 (EXTEN_CTLR2)

Offset address: 0x0C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved											IREF_INC	CC_HVT	CC3_REF	CC2_REF	CC1_REF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			ISP_NS	Reserved	ISP_BE	ISP_PS	ISP_AE	Reserved		QII_FILT	Reserved	QII_HYP	QII_AV	QII_PS	QII_AE

Bit	Name	Access	Description	Reset value
[31:21]	Reserved	RO	Reserved	0
20	IREF_INC	RW	PD PHY and BC pin reference current status selections: 1: 12% increase in PD PHY and BC pin reference current; 0: Rated PD PHY and BC pin reference current.	0
19	CC_HVT	RW	CC pin high threshold input mode enable: 1: High threshold input, which can reduce power consumption during PD communication; 0: Normal GPIO threshold input.	0
18	CC3_REF	RW	CC3 pin analog reference enable: 1: Reference enabled, recommended to be normally open; 0: Reference closed.	0
17	CC2_REF	RW	CC2 pin analog reference enable: 1: Reference enabled, recommended to be normally open; 0: Reference closed.	0
16	CC1_REF	RW	CC1 pin analog reference enable: 1: Reference enabled, recommended to be normally open; 0: Reference closed.	0
[15:13]	Reserved	RO	Reserved	0
12	ISP_NS	RW	Current sampling amplifier ISP_OP negative input channel selection: 1: The negative input terminal is grounded for single-ended input amplification; 0: The PB7 pin is used as the negative input terminal for differential input amplification.	0
11	Reserved	RO	Reserved	0
10	ISP_BE	RW	Current sampling amplifier ISP_OP self-bias enable: 1: Enable bias and increase DC bias voltage; 0: Disable bias.	0
9	ISP_PS	RW	Current sampling amplifier ISP_OP positive input channel selection: 1: PA6 pin is used as the positive input; 0: ISP pin is used as the positive input.	0
8	ISP_AE	RW	Current sampling amplifier ISP_OP enable: 1: ISP_OP enable; 0: ISP_OP disable.	0
[7:6]	Reserved	RO	Reserved	0
5	QII_FILT	RW	Signal decoding QII digital filter enable: 1: QII digital filter enabled; 0: QII digital filter disabled.	0
4	Reserved	RO	Reserved	0

3	QII_HYP	RW	Signal decoding comparator QII_CMP hysteresis voltage selection: 1: Option 2; 0: default value, rated 200mV. *Note: Please refer to the data sheet "CH641DS0" for specific data.*	0
2	QII_AV	RW	Signal decoding amplifier QII_OP gain selection: 1: Higher closed-loop gain, about 23 times; 0: Lower closed-loop gain, about 15 times.	0
1	QII_PS	RW	Signal decoding amplifier QII_OP input channel selection: 1: PA7 as input; 0: PB8 as input.	0
0	QII_AE	RW	Signal decoding amplifier QII_OP and comparator QII_CMP enable: 1: Enable the amplifier and comparator, use the output of the comparator as the input of the GPIO pin; 0: Disable the amplifier and comparator, input directly from GPIO.	0

Chapter 17 Debug Support (DBG)

17.1 Main Features

This register allows configuration of the MCU in debug state. include:

- Support counter of Window Watchdog (WWDG)
- Support counter of Timer 1
- Support counter of Timer 2

17.2 Register Description

17.2.1 Debug MCU Configuration Register (DBGMCU_CTRL)

Address: 0x7C0(CSR)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		TIM2_STOP	TIM1_STOP	Reserved		WWDG_STOP	Reserved						STANDBY	Reserved	SLEEP

Bit	Name	Access	Description	Reset value
[31:14]	Reserved	RW	Reserved	0
13	TIM2_STOP	RW	Timer 2 debug stop bit. The counter stops working when the core enters debug state. 1: The counter of timer 2 stops working; 0: The counter of timer 2 still works normally.	0
12	TIM1_STOP	RW	Timer 1 debug stop bit. The counter stops working when the core enters debug state. 1: The counter of timer 1 stops working; 0: The counter of timer 1 still works normally.	0
[10:11]	Reserved	RW	Reserved	0
9	WWDG_STOP	RW	Window watchdog debugging stop bit. The debug window watchdog stops working when the core enters debug state. 1: The window watchdog counter stops working; 0: The window watchdog counter still works normally.	0
[8:3]	Reserved	RW	Reserved	0
2	STANDBY	RW	Debug standby mode bit. 1: (FCLK on, HCLK on) The digital circuit part is not powered off, and the FCLK and HCLK clocks are provided by the internal RL oscillator. In addition, the microcontroller exits STANDBY mode by generating a system reset, which is the same as reset; 0: (FCLK off, HCLK off) The entire digital circuit section is powered off. From a software point of view, exiting STANDBY mode is the same as resetting (except that some status bits indicate that the microcontroller has just exited STANDBY state).	0
1	Reserved	RW	Reserved	0
0	SLEEP	RW	Debug sleep mode bit.	0

			1: (FCLK on, HCLK on) In sleep mode, FCLK and HCLK clocks are provided by the originally configured system clock; 0: (FCLK on, HCLK off) In sleep mode, FCLK is provided by the originally configured system clock. clock is provided and HCLK is turned off. Since sleep mode does not reset the configured clock system, software does not need to reconfigure the clock system when exiting from sleep mode.	